Spline and Spline Wavelet Methods with Applications to Signal and Image Processing

Amir Z. Averbuch · Pekka Neittaanmaki
Valery A. Zheludev

Spline and Spline Wavelet Methods with Applications to Signal and Image Processing

Volume I: Periodic Splines

Springer

Amir Z. Averbuch
Valery A. Zheludev
School of Computer Science
Tel Aviv University
Tel Aviv
Israel

Pekka Neittaanmaki
Mathematical Information Technology
University of Jyväskylä
Jyväskylä
Finland

Additional material to this book can be downloaded from http://www.extras.springer.com

ISBN 978-94-024-0562-0 ISBN 978-94-017-8926-4 (eBook)
DOI 10.1007/978-94-017-8926-4
Springer Dordrecht Heidelberg New York London

To
Dorit Peled-Averbuch
Neittaanmaki Family
Tatiana Zheludev

Preface

Since their introduction in the pioneering work by Schoenberg [75], splines have become one of the powerful tools in Mathematics [2, 47, 76, 77, 95] and in computer-aided geometric design [22, 27, 46, 48, 57, 103]. In recent decades, splines have served as a source for wavelet [1, 3, 4, 12, 14, 17, 29, 39, 40, 58, 71, 80, 88, 91, 92, 96, 100, 101, 102], multiwavelet [13, 43], and frame constructions [16, 19, 21, 37, 38, 44, 67, 72]. Splines and spline-based wavelets, wavelet packets and frames have been extensively used in signal and image processing applications [5, 6, 11, 15, 18, 23, 24, 26, 31, 32, 51, 53, 54, 65, 81, 85, 87, 89, 90], to name a few.

An excellent survey for state-of-the-art (as of year 1999) on spline theory and applications is given in [86]. This survey motivated us in writing the present textbook. Another motivation was the emergence in recent years of new contributions of splines to wavelet analysis and applications. In addition, we believe that the socalled discrete splines and their applications deserve a systematic exposure.

Discrete splines [30, 52, 60, 61, 62, 63, 68, 77, 93], whose properties mimic the properties of polynomial splines, are the discrete-time counterparts of polynomial splines. They provide natural tools for handling discrete-time signal processing problems and serve as a source for the design of wavelet transforms [9, 10, 55, 69] and frames transforms [16, 20, 105], whose properties perfectly fit signal/image processing applications.

The goal of this book is to provide a universal toolbox accompanied by a MATLAB software for manipulating polynomial and discrete splines, spline-based wavelets, wavelet packets, and wavelet frames in signal/image processing applications. For this, known and new contributions of splines to signal and image processing are described from a unified perspective, which is based on the so-called Zak transform (ZT) [28, 94]. Being applied to B-splines, the ZT produces sets of so-called exponential splines (in Schoenberg [76] sense), which are similar to Fourier exponentials. This approach provides explicit constructions of different types of splines such as interpolating, quasi-interpolating and smoothing splines, best approximation splines and orthonormal bases for spline spaces. Constructions and utilization of various spline wavelets and spline wavelet packets have become straightforward. The ZT of discrete B-splines produces exponential discrete splines.

Coupled with the Lifting scheme [82] of a wavelet transform, the ZT approach utilizes polynomial and discrete splines for the design of versatile library of

biorthogonal wavelets, multiwavelets, and wavelet frames (framelets) [9, 10, 12, 13, 14, 16, 17, 19, 20, 21, 105]. Properties of the designed wavelets and framelets, such as symmetry, flat spectra, vanishing moments, and good localization in either time or frequency domains, are valuable for signal/image processing applications. For example, the so-called Butterworth biorthogonal wavelets and wavelet frames, which originate from discrete splines, have proved to be especially efficient in signal/image processing applications. Digital filters, which have been produced during wavelets design process, give birth to subdivision schemes for fast explicit computation of splines values at dyadic and triadic rational points [22, 103, 104], which is needed for interpolation, resampling, and geometric transformations of images.

Periodic exponential splines form orthogonal bases of periodic splines spaces, which are very similar to periodic Fourier exponentials. Representation of periodic splines via orthonormal bases produces the so-called Spline Harmonic Analysis (SHA) [99, 101], which combines approximation abilities of splines with the computational strength of the Fast Fourier transform (FFT). It introduces the harmonic analysis methodology into periodic spline spaces.

SHA enables us to efficiently construct and manipulate different types of splines, wavelets, wavelet packets, and wavelet frames. SHA has paved the way for periodic splines to contribute to solutions of signal/image processing applications [12, 18, 23, 24, 26, 101, 105].

The textbook is divided into two volumes. In Volume I, periodic splines and their diverse signal processing applications are discussed. Volume II deals with nonperiodic splines.

The following topics are explored in Volume I of the book:

Zak transform (ZT): ZT of periodic functions is introduced and its properties are outlined. In particular, they include periodic counterparts of the Poisson summation formulas (for example, [70, 79]). Realizations of the ZT in polynomial and discrete periodic spline spaces result in the SHA, which is presented in detail.

Elements of spline theory and design: Different types of periodic polynomial and discrete splines with equidistant nodes are presented and their properties are outlined. The design of interpolating, smoothing, shift-orthogonal splines becomes straightforward due to the SHA methodology. Constructions of these splines types utilize filters with infinite impulse response (IIR). However, due to the FFT utilization, the computational cost of filtering with IIR is similar to the cost of filtering with finite impulse response (FIR) filters. FIR filtering enables real-time processing. For this purpose, the so-called local quasi-interpolating and smoothing splines, which are constructed by filtering data samples with FIR filters, can be used, [17, 97, 98]. Their properties are close to the properties of global interpolating and smoothing splines. These splines are presented in details in Volume II of this book. However, a couple of examples of quasi-interpolating splines are given in the current volume.

Spline subdivision and signals (images) upsampling: If spline values at grid points are given, the computation of its values between the grid points is called a spline subdivision. We describe SHA-based fast subdivision algorithms, which

explicitly derive spline values at dyadic and triadic rational points from the samples taken at integer grid points in one and two dimensions. The computer-aided geometric design is a main field of application for subdivision schemes (for example, see [27, 73]). However, these techniques suit well for signals and images upsampling, to restore sparsely sampled signals and images at intermediate points. These upsampling procedures increase the objects resolution. On the other hand, when data are corrupted by noise, upsampling from a sparse grid can significantly reduce the noise level.

Deconvolution: Deconvolution here means restoration of a signal or an image from blurred sampled data that are typically corrupted by noise. This is an ill-posed problem, which means that even small fluctuations in the input may lead to output instability. Fourier analysis is a good tool in handling the convolution and convolution-related problems, such as inverse of the heat equation and the Cauchy problem for the Laplace equation, to name a few. This is because that Fourier exponentials are the eigenfunctions of the convolution operator. The exponential splines possess a similar property.

Many deconvolution algorithms are based on the Tikhonov regularization scheme [83, 84], where the approximated solution of the deconvolution problem is reduced to minimizing a parameterized functional. The regularization parameter provides a tradeoff between the approximation of the available data and the regularity of the solution. Typically, data are discrete and corrupted by noise but the physical meaning of the problem dictates the solution smoothness. In these setups, splines are a good match. SHA provides a unified computational scheme for finding a stable solution that possesses the required smoothness. The regularization parameter is derived automatically from the evaluation of the relative contributions of the coherent signal and the noise in the data [18]. Note that construction of a smoothing spline can be regarded as a special case of the regularization algorithm when the data are discrete and noised and blurring is not present.

Design of spline wavelets and wavelet packets: Constructions of different types of spline wavelets and wavelet packets in an explicit form and fast implementation of the corresponding transforms using FFT are described. The spline wavelets construction stems from the two-scale relations between exponential splines from different resolution scales, which are simple and the computation complexity of which is practically independent of the splines order. The Fourier spectra of spline wavelets from different resolution scales partitions the frequency domain in a logarithmic way. Spline wavelet packets, which generalize wavelets, are designed by using the same two-scale relations. The Fourier spectra of spline wavelet packets from the j-th resolution scale split the frequency domain into 2^j bands of approximately equal width. The shapes of the magnitude spectra of the spline wavelet packets tend to be rectangular as the splines order increases. Therefore, the wavelet packet transform of a signal produces a number of the signal representations, which are associated with different sets of frequency bands. An optimal (up to a certain cost function) representation is achieved by the so-called Best Basis algorithm [41, 74]. On the other hand, spline wavelet packets provide a versatile collection of waveforms with different spans and different

frequency contents. This collection can be used as a dictionary for the Matching Pursuit framework [45, 59, 66].

Deconvolution revisited: When a signal/image is restored from blurred sampled data affected by noise, it is worthwhile to keep in mind that relative contributions of a coherent signal and noise are different in different frequency components of the data. The deconvolution problem then can be solved separately in different frequency bands, while regularization parameters are found according to the signal-to-noise ratio in each band. This approach significantly extends the adaptation abilities and the method robustness. Practically, this scheme is implemented via the utilization of orthonormal spline wavelet packets. SHA provides a unified computational scheme for their design, a fast implementation of the algorithm and an explicit representation of the solutions. An optimal set of wavelet packets is selected by the application of the Best Basis algorithm. Then, the equation is solved separately for each wavelet packet transform coefficients block [24].

If an original signal to be restored from convolution with a bandlimited kernel is highly inhomogeneous and the discrete output is strongly noised, a method, which can be characterized as a Regularized Matching Pursuit (RMP) method, produces good results. RMP is a greedy algorithm, which uses orthonormal spline wavelet packet dictionaries. The main distinction from the conventional Matching Pursuit is utilization of two different dictionaries. One dictionary, which consists of discrete-time signals, is used to test discrete data. The second dictionary, which consists of spline wavelet packets, approximates the continuous solution.

The regularization is achieved by replacing the orthogonal projections of the data onto the dictionary elements, which are used in the conventional Matching Pursuit, by oblique projections.

The above-listed regularized deconvolution methods turn into denoising methods when the input data is not blurred.

Acoustic classification and target detection: The spline wavelet packet transforms, which produce multiple splits of the frequency domain of a signal, can be used to reveal frequency bands, which are characteristic either for the signal or for a class of signals. Determining such bands is important when the processing goal is to detect the presence or arrival of objects of a certain type or to classify objects of different types via the analysis of their acoustic signatures. Typically, signals emitted by mechanical objects have a quasi-periodic structure. Such acoustic signals contain only a few dominating bands in the frequency domain, whose general disposition remains stable with respect to varying conditions. The combination of the inherent energies in these frequency bands can form an acoustic signature of the object. An efficient way to determine the characteristic bands is to apply an orthogonal spline wavelet packet transform to the signal and calculate energies in the blocks of the transform coefficients [7, 25, 26]. This approach is illustrated in the book on detection of the arrival of a vessel of a certain type by analyzing hydro-acoustic signals recorded by a hydrophone. However, the above approach is applicable to processing other types of quasi-periodic signals such as biomedical signals [8].

Discrete splines: The space of periodic discrete splines, which is a subspace of the periodic signals space is described. Properties of the discrete splines mirror properties of polynomial splines. The Zak transform applied to the discrete B-splines produces exponential discrete splines, which serve as a source for the construction of a discrete version of the SHA [68, 69]. The discrete SHA simplifies manipulations with the discrete splines. In particular, it provides explicit expressions for interpolating and smoothing discrete splines in one and two dimensions and provides fast algorithms for calculation of the discrete splines values from grid samples. This is a useful tool for upsampling signals and images.

Discrete splines wavelets and wavelet packets: Similarly to the polynomial splines case, the wavelet and wavelet packet transforms are introduced to the discrete splines space. These transforms are based on the relations between the exponential discrete splines from different resolution scales. Practically, the transforms of periodic signals are implemented by multirate filtering signals by two-channel filter banks with the downsampling factor of 2 (critically sampled filter banks). The filtering implementation is accelerated by switching to the polyphase representation of signals and filters and using the FFT. The field of applications of the discrete splines wavelet and wavelet packet transforms is, generally, the same as the field of their polynomial counterparts. In particular, the discrete splines version of the RMP is outlined.

Design of biorthogonal wavelets: The polynomial and discrete splines may contribute to wavelet analysis in another way. They are a source for a family of filters, which generate biorthogonal wavelets, whose properties are valuable for signal processing. Although these wavelets originate from splines, they, unlike the spline wavelets, do not belong to spline spaces. Design of biorthogonal wavelets and efficient implementation of the transforms of signals is carried out through the so-called Lifting scheme [82]. The idea is to split the signal into even and odd subarrays. Then, the even subarray is filtered using some *prediction* filter in order to predict the odd subarray. The predicted subarray is extracted from the original odd subarray. The difference array is filtered by an *update* filter and it is used to update the even subarray in order to eliminate aliasing. These operations are then applied to the updated even subarray and so on. Thus, multiscale wavelet decomposition is achieved. Reconstruction is implemented in the reverse order. The key point is a proper choice of the prediction and update filters. Naturally, odd samples can be predicted from midpoint values of either polynomial or discrete splines, which interpolate or quasi-interpolate the even samples of the signal. In this way, a number of linear phase IIR and FIR prediction filters are designed. Being properly modified, they are used for the update step as well. By using these filters, a diverse library of biorthogonal wavelets is constructed [9, 12]. Exclusive properties are demonstrated by the so-called Butterworth wavelets, which originate from discrete interpolatory splines. They are related to the Butterworth filters [64], which are widely used in signal processing.

Data compression is an important application of the wavelet analysis. The spline-based biorthogonal wavelets proved to be highly efficient in achieving

highquality low-bitrate compression. This application is discussed in detail in Volume II.

Wavelet frames (framelets): Recently frames or redundant expansions of signals have attracted considerable interest from researchers working in signal processing although one particular class of frames, the Gabor systems, has been applied and investigated since 1946 [49]. As the requirement of one-to-one correspondence between the signal and its transform coefficients is dropped, there is more freedom to design and implement frame transforms.

Here we present some key features of framelet transforms and application to image restoration.

Design and implementation: Wavelet transforms use critically sampled filter banks. On the other hand, wavelet frame transforms are implemented by application of oversampled perfect reconstruction (PR) pairs of filter banks. It means that the number of channels in the filter banks exceeds the downsampling factor. Moreover, translations of the filters, that constitute such filter banks, form wavelet frames in the signal space [33, 42, 56]. Generally, the synthesis filter bank in the PR pair differs from the analysis filter bank. In the case when both filter banks are the same, the corresponding frame is tight. Tight frames can be regarded as redundant counterparts of orthogonal bases. The design of a variety of three- and four-channel PR filter banks, which generate tight frames in the space of periodic signals, is described. The filter banks comprise one low-pass, one high-pass and either one or two band-pass filters. All these filters are derived from the spline-based prediction filters, which were used for the design of biorthogonal wavelet transforms. In addition, the so-called semi-tight frames are introduced, where the low- and high-pass filters in the synthesis filter bank are the same as in the analysis filter bank, while the bandpass filters are different. Periodic setting, which enables the utilization of a wide range of IIR filter banks with a relaxation of the tightness requirement, provides a number of additional opportunities. Properties such as symmetry, interpolation, flat spectra combined with fine time-domain localization of framelets as well as a high number of vanishing moments can be easily achieved. The transforms implementation is reduced to application of the direct and the inverse FFT.

Image restoration: A valuable advantage that redundant representations hold is their ability to restore missing and incomplete information, which is based on the prior assumption that a frame expansion of a given signal/image is sparse. In principle, only part of the samples/pixels is needed to (near) perfect object restoration. This approach, which is a variation of the *Compressive Sensing* methodology (see, for example, [36]), has proved to be extremely efficient for image restoration.

In practice, this approach is implemented via minimization of a parameterized functional where the sparse representation is reflected in the $l1$ norm of the frame transform coefficients. The $\|\cdot\|_1$ minimization does not have an explicit solution and can be resolved only by iterative methods. The so-called *split Bregman iteration* (SBI) scheme [50] provides a fast and stable algorithm for that. Variations of

this scheme and its application to image restoration using wavelet frames are described in [34, 35], to mention a few. A variety of impressive results on image restoration have been reported in the last couple of years. A survey is given in [78] while a recent development is described in [34].

Due to applications diversity, it is important to have a library of wavelet frames in order to select a frame that fits best to a specific task. Forward and inverse transforms in iterative algorithms are repeated many times: therefore, members in this library must have fast and stable transforms implementation. Waveforms symmetry with the availability of vanishing moments is also important in order to avoid distortions when thresholding is used.

The designed family of the spline-based wavelet frames perfectly meet these requirements. A number of experiments on image restoration, where performance of different frames is compared with each other, is described. Their diversity enabled a frame to be selected that best fits to each specific application. In particular, in most of the experiments semi-tight frames outperform tight frames.

The framelets construction is presented together with a fast computational scheme in [19–21].

All the presented methods are accompanied by Matlab codes. A guide to the software is given in Appendix.

The authors thank Steve Legrand for thorough proofreading and his comments, which resulted in many corrections of the text.

Tel Aviv, Israel, November 2013 — Amir Z. Averbuch
Valery A. Zheludev
Jyväskylä, Finland — Pekka Neittaanmaki

References

1. P. Abry, A. Aldroubi, Designing multiresolution analysis-type wavelets and their fast algorithms. J. Fourier Anal. Appl. **2**(2), 135–159 (1995)
2. J.H. Ahlberg, E.N. Nilson, J.L. Walsh, *The Theory of Splines and Their Applications* (Academic Press, New York, 1987)
3. A. Aldroubi, M. Eden, M. Unser, Discrete spline filters for multiresolutions and wavelets of l_2. SIAM J. Math. Anal. **25**(5), 1412–1432 (1994)
4. A. Aldroubi, M. Unser, Families of multiresolution and wavelet spaces with optimal properties. Numer. Funct. Anal. Optim. **14**(5–6), 417–446 (1993)
5. O. Amrani et al., Symmetric interpolatory framelets and their erasure recovery properties. Int. J. Wavelets Multiresolut. Inf. Process. **5**(4), 541–566 (2007)
6. M. Antonini et al., Image coding using wavelet transform. IEEE Trans. Image Process. **1**(2), 205–220 (1992)
7. A. Averbuch et al., A wavelet packet algorithm for classification and detection of moving vehicles. Multidimens. Syst. Signal Process. **12**(1), 9–31 (2001)
8. A. Averbuch, I. Kozlov, V. Zheludev, Wavelet-packet-based algorithm for identification of quasi-periodic signals, ed. by A.F. Laine, M.A. Unser, A. Aldroubi. Wavelets: Applications in Signal and Image Processing IX, vol. 4478 of Proc. SPIE, 353–360 (2001)

9. A. Averbuch, A.B. Pevnyi, V. Zheludev, Biorthogonal Butterworth wavelets derived from discrete interpolatory splines. IEEE Trans. Signal Process. **49**(11), 2682–2692 (2001)
10. A. Averbuch, A.B. Pevnyi, V. Zheludev, Butterworth wavelet transforms derived from discrete interpolatory splines: recursive implementation. Signal Process. **81**(11), 2363–2382 (2001)
11. A. Averbuch, V. Zheludev, Image compression using spline based wavelet transforms ed. by A. Petrosian, F. Meyer, in *Wavelets in Signal and Image Analysis: From Theory to Practice*, (Kluwer Academic Publishers, Dordrecht, 2001) p. 341–376
12. A. Averbuch, V. Zheludev, Construction of biorthogonal discrete wavelet transforms using interpolatory splines. Appl. Comput. Harmon. Anal. **12**(1), 25–56 (2002)
13. A. Averbuch, V. Zheludev, Lifting scheme for biorthogonal multiwavelets originated from Hermite splines. IEEE Trans. Signal Process. **50**(3), 487– 500 (2002)
14. A. Averbuch, V. Zheludev, Splines: a new contribution to wavelet analysis. ed. by J. Levesley, I.J. Anderson, J.C. Mason, Algorithms for Approximation IV: in *Proceedings of the 2001 International Symposiu*m, (University of Huddersfield, 2002) p. 314–321
15. A. Averbuch, V. Zheludev, A new family of spline-based biorthogonal wavelet transforms and their application to image compression. IEEE Trans. Image Process. **13**(7), 993–1007 (2004)
16. A. Averbuch, V. Zheludev, Wavelet and frame transforms originated from continuous and discrete splines. ed. by J. Astola, L. Yaroslavsky, Advances in Signal Transforms: Theory and Applications, (Hindawi Publishing Corporation, New York, 2007) p. 1–56
17. A. Averbuch, V. Zheludev, Wavelet transforms generated by splines. Int. J. Wavelets Multiresolut. Inf. Process. **5**(2), 257–291 (2007)
18. A. Averbuch, V. Zheludev, Spline-based deconvolution. Signal Process. **89**(9), 1782–1797 (2009)
19. A. Averbuch, V. Zheludev, T. Cohen, Interpolatory frames in signal space. IEEE Trans. Signal Process. **54**(6), 2126–2139 (2006)
20. A. Averbuch, V. Zheludev, T. Cohen, Tight and sibling frames originated from discrete splines. Signal Process. **86**(7), 1632–1647 (2006)
21. A. Averbuch, V. Zheludev, T. Cohen, Multiwavelet frames in signal space originated from Hermite splines. IEEE Trans. Signal Process. **55**(3), 797–808 (2007)
22. A. Averbuch et al., Ternary interpolatory subdivision schemes originated from splines. Int. J. Wavelets Multiresolut. Inf. Process. **9**(4), 611–633 (2011)
23. A. Averbuch, V. Zheludev, M. Khazanovsky, Deconvolution by matching pursuit using spline wavelet packets dictionaries. Appl. Comput. Harmon. Anal. **31**(1), 98–124 (2011)
24. A. Averbuch et al., Block based deconvolution algorithm using spline wavelet packets. J. Math. Imaging Vision. **38**(3), 197–225 (2010)
25. A. Averbuch et al., Acoustic detection and classification of river boats. Appl. Acoustics **72**(1), 22–34 (2011)
26. A. Averbuch et al., Wavelet-based acoustic detection of moving vehicles. Multidimens. Syst. Signal Process. **20**(1), 55–80 (2009)
27. R. Bartels, J. Beatty, B. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling* (Morgan Kaufmann Publishers, San Mateo, CA, 1987)
28. M.J. Bastiaans, Gabor's Expansion and the Zak Transform for Continuoustime and Discrete-Time Signals, ed. by Y.Y. Zeevi, R. Coifman, Signal and Image Representation in Combined Spaces, number 7 in Wavelet Anal. Appl. (Academic Press, San Diego, CA, 1998) pp. 23–69
29. G. Battle, A block spin construction of ondelettes. I. lemarié functions. Comm. Math. Phys. **110**(4), 601–615 (1987)
30. M.G. Ber, Natural discrete splines and the averaging problem. Vestnik Leningrad Univ. Math. **23**(4), 1–4 (1990)

31. O. Bernard et al., Variational B-spline level-set method for fast image segmentation in 2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro. Proceedings, p. 177–180, IEEE, 2008
32. T. Blu et al., Sparse sampling of signal innovations. IEEE Signal Process. Mag. **25**(2), 31–40 (2008)
33. H. Bölcskei, F. Hlawatsch, H.G. Feichtinger, Frame-theoretic analysis of oversampled filter banks. IEEE Trans. Signal Process. **46**(12), 3256–3268 (1998)
34. J. Cai et al., Image restoration: total variation, wavelet frames, and beyond. J. Amer. Math. Soc. **25**(4), 1033–1089 (2012)
35. J. Cai, S. Osher, Z. Shen, Split Bregman methods and frame based image restoration. Multiscale Model. Simul. **8**(2), 337–369 (2009/2010)
36. E. Candes et al., Stable signal recovery from incomplete and inaccurate measurements. Comm. Pure Appl. Math. **59**(8), 1207–1223 (2006)
37. C.K. Chui, W. He. Compactly supported tight frames associated with refinable functions. Appl. Comput. Harmon. Anal. **8**(3), 293–319 (2000)
38. C.K. Chui, W. He, J. Stöckler, Compactly supported tight and sibling frames with maximum vanishing moments. Appl. Comput. Harmon. Anal. **13**(3), 224–262 (2002)
39. C.K. Chui, J.-Z. Wang, On compactly supported spline wavelets and a duality principle. Trans. Amer. Math. Soc. **330**(2), 903–915 (1992)
40. A. Cohen, I. Daubechies, J.-C. Feauveau, Biorthogonal bases of compactly supported wavelets. Comm. Pure Appl. Math. **45**(5), 485–560 (1992)
41. R.R. Coifman, V.M. Wickerhauser, Entropy-based algorithms for best basis selection. IEEE Trans. Inform. Theory **38**(2), 713–718 (1992)
42. Z. Cvetković, M. Vetterli, Oversampled filter banks. IEEE Trans. Signal Process. **46**(5), 1245–1255 (1998)
43. W. Dahmen et al., Biorthogonal multiwavelets on the interval: Cubic Hermite splines. Constr. Approx. **16**(2), 221–259 (2000)
44. I. Daubechies et al., Framelets: MRA-based constructions of wavelet frames. Appl. Comput. Harmon. Anal. **14**(1), 1–46 (2003)
45. G. Davis, S. Mallat, Z. Zhang, Adaptive time-frequency decompositions with matching pursuits. SPIE J. Opt. Eng. **33**(7), 2183–2191 (1994)
46. P. Davis, B-splines and geometric design. SIAM News **29**(5) (1997)
47. C. de Boor, *A Practical Guide to Splines* (Springer, New York, 1978)
48. P. Dierckx, *Curves and Surface Fitting with Splines* (Oxford University Press, New York, 1993)
49. D. Gabor, Theory of communications. J. Inst. Electr. Eng. **93**, 429–457 (1946)
50. T. Goldstein, S. Osher, The split Bregman method for L1-regularized problems. SIAM J. Imaging Sci. **2**(2), 323–343 (2009)
51. H.S. Hou, H. C. Andrews, Cubic splines for image interpolation and digital filtering. IEEE Trans. Acoust. Speech Signal Process. **26**(6), 508–517 (1978)
52. K. Ichige, M. Kamada, An approximation for discrete B-splines in time domain. IEEE Signal Process. Lett. **4**(3), 82–84 (1997)
53. S. Jonić et al., An optimized spline-based registration of a 3D CT to a set of C-arm images. Internat. J. Biomed. Imaging 2006(47197), p. 12 (2006)
54. M. Kamada, K. Toraichi, R.E. Kalman, A smooth signal generator based on quadratic B-spline functions. IEEE Trans. Signal Process. **43**(5), 1252–1255 (1995)
55. V.A. Kirushev, V.N. Malozemov, A.B. Pevnyi, Wavelet decomposition of the space of discrete periodic splines. Math. Notes **67**(5–6), 603–610 (2000)
56. J. Kovacevic, P.L. Dragotti, V.K. Goyal, Filter bank frame expansions with erasures. IEEE Trans. Inform. Theory **48**(6), 1439–1450 (2002)
57. B. Kvasov, *Methods of Shape-Preserving Spline Approximation* (World Scientific, River Edge, NJ, 2000)

58. P.G. Lemarié. Ondelettes à localisation exponentielle. J. Math. Pures Appl. (9), **67**(3), 227–236 (1988)
59. S. Mallat, Z. Zhang, Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. **41**(12), 3397–3415 (1993)
60. V.N. Malozemov, N.V. Chashnikov, Discrete periodic splines with vector coefficients and geometric modeling. Doklady Math. **80**(3), 797–799 (2009)
61. V.N. Malozemov, A.B. Pevnyi, Discrete periodic B-splines. Vestn St. Petersburg Univ. Math. **30**(4), 10–14 (1997)
62. V.N. Malozemov, A.B. Pevnyi, Discrete periodic splines and their numerical applications. Comput. Math. Math. Phys. **38**(8), 1181–1192 (1998)
63. V.N. Malozemov, A.N. Sergeev, Discrete nonperiodic splines on a uniform grid, ed. by N.N. Uraltseva, in *Proceedings of the St. Petersburg Mathematical Society, Vol. VIII*, vol. 205 of Amer. Math. Soc. Transl. Ser. 2, p. 175–187 (2002) First published in Trudy Sankt-Peterburgskogo Matematicheskogo Obshchestva, Vol. 8 (2000), 199–213 (Russian)
64. A.V. Oppenheim, R.W. Schafer, *Discrete-time signal processing*, 3rd edn. (Prentice Hall, New York, 2010)
65. D. Pang, L.A. Ferrari, P.V. Sankar, B-spline FIR filters. Circ. Syst. Sig. Proc. **13**(1), 31–64 (1994)
66. Y.C. Pati, R. Rezaiifar, P.S. Krishnaprasad, Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition, in *Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems & Computers*, vol. 1, p. 40–44, (1993)
67. A.P. Petukhov, Symmetric framelets. Constr. Approx. **19**(2), 309–328 (2003)
68. A.B. Pevnyi, V. Zheludev. On the interpolation by discrete splines with equidistant nodes. J. Approx. Theory **102**(2), 286–301 (2000)
69. A.B. Pevnyi, V. Zheludev. Construction of wavelet analysis in the space of discrete splines using Zak transform. J. Fourier Anal. Appl. **8**(1), 59–83 (2002)
70. M. Pinsky, *Introduction to Fourier Analysis and Wavelets* (Brooks Cole, 2002)
71. G. Plonka, M. Tasche, On the computation of periodic spline wavelets. Appl. Comput. Harmon. Anal. **2**(1), 1–14 (1995)
72. A. Ron, Z. Shen, Compactly supported tight affine spline frames in $L^2\mathbb{R}^d$. Math. Comp. **67**(221), 191–207 (1998)
73. M. Sabin, *Analysis and Design of Univariate Subdivision Schemes, Geometry and Computing*, vol. 6 (Springer, Berlin, 2010)
74. N. Saito, R.R. Coifman, Local discriminant bases and their applications. J. Math. Imaging Vision **5**(4), 337–358 (1995)
75. I.J. Schoenberg, Contributions to the problem of approximation of equidistant data by analytic functions. Quart. Appl. Math. **4**, 45–99, 112–141 (1946) Parts A and B
76. I.J. Schoenberg, *Cardinal Spline Interpolation* (SIAM, Philadelphia, PA, 1973)
77. L.L. Schumaker, *Spline Functions: Basic Theory* (Wiley, New York, 1981)
78. Z. Shen, Wavelet frames and image restorations. in *Proceedings of the International Congress of Mathematicians*, ed. by R. Bhatia, vol. 6. (Hindustan Book Agency, New Delhi, 2010), p. 2834–2863
79. E. Stein, G. Weiss. *Introduction to Fourier Analysis on Euclidean Spaces* (Princeton University Press, Princeton, 1971)
80. J.-O. Strömberg, A modified Franklin system and higher-order spline systems of Rn as unconditional bases for Hardy spaces, in *Conference on Harmonic Analysis in Honor of Antoni Zygmund,* vol. 1, 2 (Chicago, Ill., 1981) (Wadsworth, Belmont, CA, 1983) pp. 475–494
81. H. Sugiyama, M. Kamada, R. Enkhbat. Image compression by spline wavelets orthogonal with respect to weighted sobolev inner product. Advanced Modeling and Optimization **10**(2), 163–175 (2008)

82. W. Sweldens, The lifting scheme: a custom-design construction of biorthogonal wavelets. Appl. Comput. Harmon. Anal. **3**(2), 186–200 (1996)
83. A.N. Tikhonov, Solution of incorrectly formulated problems and the regularization method. Soviet Math. Dokl. **4**(4), 1035–1038 (1963)
84. A.N. Tikhonov, V.Y. Arsenin, *Solutions of Ill-Posed Problems* (Wiley, New York, 1977)
85. K. Toraichi et al., Two-dimensional spline interpolation for image reconstruction. Pattern Recognit. **21**(3), 275–284 (1988)
86. M. Unser, Splines: a perfect fit for signal and image processing. IEEE Signal Process. Mag. **16**(6), 22–38 (1999) IEEE Signal Processing Society's 2000 magazine award
87. M. Unser, Splines: in A perfect fit for medical imaging, ed. by M. Sonka, J.M. Fitzpatrick, Medical Imaging 2002: Image Processing, of Proc. SPIE, vol. 4684 (2002) p. 225–236
88. M. Unser, A. Aldroubi, M. Eden, On the asymptotic convergence of Bspline wavelets to Gabor functions. IEEE Trans. Inform. Theory **38**(2) 864–872 (1992)
89. M. Unser, A. Aldroubi, M. Eden, B-spline signal processing. I. Theory. IEEE Trans. Signal Process. **41**(2), 821–833 (1993) IEEE Signal Processing Society's 1995 best paper award
90. M. Unser, A. Aldroubi, M. Eden, B-spline signal processing. II. Efficiency design and applications. IEEE Trans. Signal Process. **41**(2), 834–848, (1993)
91. M. Unser, A. Aldroubi, M. Eden. A family of polynomial spline wavelet transforms. Signal Process. **30**(2), 141–162 (1993)
92. M. Unser, P. Thévenaz, A. Aldroubi, Shift-orthogonal wavelet bases using splines. IEEE Signal Process. Lett. **3**(3), 85–88 (1996)
93. K.F. Üstüner, L.A. Ferrari, Discrete splines and spline filters. IEEE Trans. Circuits Syst. II: Analog Digital Signal Process. **39**(7), 417–422 (1992)
94. J. Zak, Finite translations in solid-state physics. Phys. Rev. Lett. **19**(24), 1385–1387 (1967)
95. Yu.S. Zav'yalov, B.I. Kvasov, V.L. Miroshnigenko, *Methods of Splinefunctions* (Nauka, Moscow, 1980) (In Russian)
96. V. Zheludev, in *Spline harmonic analysis and wavelet bases* ed. by W. Gautcshi, Proc. Symp. Appl. Math.
97. V. Zheludev, Local spline approximation on a uniform mesh. U.S.S.R. Comput. Math. Math. Phys. **27**(5), 8–19 (1987)
98. V. Zheludev, Local smoothing splines with a regularizing parameter. Comput. Math. Math Phys. **31**, 11–25 (1991)
99. V. Zheludev, An operational calculus connected with periodic splines. Soviet. Math. Dokl. **42**(1), 162–167 (1991)
100. V. Zheludev, Wavelets based on periodic splines. Russian Acad. Sci. Dokl. Math. **49**(2), 216–222 (1994)
101. V. Zheludev, in *Periodic splines, harmonic analysis, and wavelets* ed. by Y.Y. Zeevi, R. Coifman, Signal and Image Representation in Combined Spaces, Wavelet Anal. Appl. vol. 7 (Academic Press, San Diego, CA, 1998) p. 477–509
102. V. Zheludev, Wavelet analysis in spaces of slowly growing splines via integral representation. Real Anal. Exchange **24**(1), 229–261 (1998/1999)
103. V. Zheludev, Interpolatory subdivision schemes with infinite masks originated from splines. Adv. Comput. Math. **25**(4), 475–506 (2006)
104. V. Zheludev, A. Averbuch, Computation of interpolatory splines via triadic subdivision. Adv. Comput. Math. **32**(1), 63–72 (2010)
105. V. Zheludev, V.N. Malozemov, A.B. Pevnyi, Filter banks and frames in the discrete periodic case ed. by N. N. Uraltseva in *Proceedings of the St. Petersburg Mathematical Society, Vol. XIV*, of Amer. Math. Soc. Transl., Ser. 2, vol. 228 (2009) pp. 1–11

Contents

Chapter 1
Introduction: Signals and Transforms

Abstract In this chapter we outline some well known facts about periodic signals and transforms, which are needed throughout the book. For details we refer to the classical textbook Oppenheim and Schafer [2].

1.1 Continuous-Time Periodic Signals

In this section, properties of one- and two-dimensional periodic signals and their expansions into the Fourier series are outlined.

1.1.1 One-Dimensional Periodic Signals

Piece-wise continuous functions, which are T-periodic, such that for all $t \in \mathbb{R}$, $f(t+T) = f(t)$, are referred to as continuous-time periodic signals. The space of such signals, where the inner product and the norm are defined as

$$\langle f, h\rangle \stackrel{\text{def}}{=} \int_0^T f(t)\, h^*(t)\, dt, \quad \|f\| \stackrel{\text{def}}{=} \sqrt{\int_0^T |f(t)|^2\, dt},$$

is denoted by $\mathscr{P}_T$.

$\cdot^*$ means complex conjugation.

Note that any signal $f(t)$ of limited duration L can be embedded into a space $\mathscr{P}_T$, $T \geq L$, by periodization:

$$\tilde{f}(t) = \sum_{l\in\mathbb{Z}} f(t + lT). \tag{1.1}$$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_1,

The circular convolution of two periodic signals is

$$g = f \circledast h \Longleftrightarrow g(t) = \int_0^T f(t-\tau)\, h(\tau)\, d\tau. \tag{1.2}$$

Expansion of a periodic signal into the Fourier series is:

$$f(t) = \frac{1}{T} \sum_{n\in\mathbb{Z}} c_n(f)\, e^{2\pi i\, nt/T}, \quad c_n(f) = \int_0^T f(t)\, e^{-2\pi i\, nt/T}\, dt, \tag{1.3}$$

where $\{c_n(f)\}$ are the Fourier coefficients of the signal $f(t)$. Here is a list of a few properties of the Fourier exponentials $\left\{e^{2\pi i\, nt/T}\right\}_{n\in\mathbb{Z}}$.

Orthogonality:

$$\int_0^T e^{2\pi i\, nt/T}\, e^{-2\pi i\, lt/T} dt = T\delta[n-l], \text{ where } \delta[n] \stackrel{\text{def}}{=} \begin{cases} 1, \text{ if } n = 0; \\ 0, \text{ otherwise} \end{cases} \tag{1.4}$$

means the Kronecker delta (an impulse in signal processing). Thus, the functions $\left\{e^{2\pi i\, nt/T}\right\}_{n\in\mathbb{Z}}$ form an orthogonal basis of the space $\mathscr{P}_T$.

Shift: The Fourier exponentials are the eigenfunctions of the shift operator:

$$e^{2\pi i\, n(t+d)/T} = e^{2\pi i\, nd/T}\, e^{2\pi i\, nt/T}. \tag{1.5}$$

Circular convolution:

$$\int_0^T e^{2\pi i\, n(t-\tau)/T}\, e^{2\pi i\, l\tau/T} d\tau = T\delta[n-l]\, e^{2\pi i\, nt/T}. \tag{1.6}$$

Differentiation: The Fourier exponentials are the eigenfunctions of the differentiation operator:

$$\left(e^{2\pi i\, nt/T}\right)^{(s)} = \left(\frac{2\pi i\, n}{T}\right)^s e^{2\pi i\, nt/T}. \tag{1.7}$$

Properties (1.4)–(1.7) lead to the following relations between signals from $\mathscr{P}_T$ and their Fourier coefficients:

Parseval identities:

$$\langle f, h\rangle = \frac{1}{T} \sum_{n\in\mathbb{Z}} c_n(f)\, c_n^*(h), \quad \|f\|^2 = \frac{1}{T} \sum_{n\in\mathbb{Z}} |c_n(f)|^2. \tag{1.8}$$

Circular convolution: If $g = f \circledast h$ then

$$c_n(g) = c_n(f)\, c_n(h). \tag{1.9}$$

Shift:

$$f_d(t) \stackrel{\text{def}}{=} f(t+d) \Longleftrightarrow c_n(f_d) = e^{2\pi i\, nd/T}\, c_n(f). \tag{1.10}$$

Differentiation:

$$c_n(f^{(s)}) = \left(\frac{2\pi i\, n}{T}\right)^s c_n(f). \tag{1.11}$$

Finite differences: The functions $\Delta_d[f](t) \stackrel{\text{def}}{=} f(t+d) - f(t)$ and $\delta_d[f](t) \stackrel{\text{def}}{=} f(t+d/2) - f(t-d/2)$ are called the *forward* and the *central* finite differences of the signal $f(t)$, respectively. Their Fourier coefficients, which are derived directly from Eq. (1.10), are

$$c_n\left(\Delta_d[f]\right) = (e^{2\pi ind/T} - 1)\, c_n(f), \quad c_n\left(\delta_d[f]\right) = \left(2i \sin\frac{\pi nd}{T}\right) c_n(f). \tag{1.12}$$

Finite differences of higher order of a periodic signal $f(t)$ are defined iteratively:

$$\Delta_d^n[f](t) \stackrel{\text{def}}{=} \Delta_d[\Delta_d^{n-1}[f](t)](t), \quad \delta_d^n[f](t) \stackrel{\text{def}}{=} \delta_d[\delta_d^{n-1}[f](t)](t).$$

They are periodic signals and, their Fourier coefficients are:

$$\begin{aligned} c_n\left(\Delta_d^m[f]\right) &= (e^{2\pi ind/T} - 1)^m\, c_n(f), \\ c_n\left(\delta_d^m[f]\right) &= \left(2i \sin\frac{\pi nd}{T}\right)^m c_n(f). \end{aligned} \tag{1.13}$$

The explicit expressions for the higher order differences is

$$\Delta_d^n[f](t) = \sum_{k=0}^{n} \binom{n}{k} (-1)^{n-k} f(t+k), \tag{1.14}$$

where $\binom{n}{k}$ is the binomial coefficient. It is a direct consequence of (1.13).

1.1.2 Two-Dimensional Periodic Signals

Two-dimensional periodic signals with periods X and Y are (piece-wise) continuous functions $f(x, y)$ such that for all $(x, y) \in \mathbb{R}^2$ the identity $f(x+X, y+Y) = f(x, y)$ holds. The space of such signals is denoted by $\mathscr{P}^2_{X,Y}$. If $Y = X = T$ we say that the signal $f(x, y)$ is T-periodic and denote the corresponding space as $\mathscr{P}^2_T$. The inner product and the norm in $\mathscr{P}^2_{X,Y}$ are defined as follows:

$$\langle f, h\rangle \stackrel{\text{def}}{=} \int_0^X \int_0^Y f(x,y)\, h^*(x,y)\, dx\, dy, \quad \|f\| \stackrel{\text{def}}{=} \sqrt{\int_0^X \int_0^Y |f(x,y)|^2\, dx\, dy}. \tag{1.15}$$

The circular convolution of two periodic signals is

$$g = f \circledast h \Longleftrightarrow g(x,y) = \int_0^X \int_0^Y f(x-\xi, y-\eta)\, h(\xi,\eta)\, d\xi\, d\eta. \tag{1.16}$$

The Fourier expansion of a periodic signal is:

$$\begin{aligned} f(x,y) &= \frac{1}{XY} \sum_{k,l\in\mathbb{Z}} c_{k,l}(f)\, e^{2\pi i(kx/X + ly/Y)}, \\ c_{k,l}(f) &= \int_0^X \int_0^Y f(x,y)\, e^{-2\pi i(kx/X+ly/Y)}\, dx\, dy. \end{aligned} \tag{1.17}$$

Relations between 2D periodic signals and their Fourier coefficients are similar to respective relations of the 1D periodic signals. In particular:

Parseval identities:

$$\langle f, h\rangle = \frac{1}{XY} \sum_{k,l\in\mathbb{Z}} c_{k,l}(f)\, c^*_{k,l}(h), \quad \|f\|^2 = \frac{1}{XY} \sum_{k,l\in\mathbb{Z}} |c_{k,l}(f)|^2. \tag{1.18}$$

Circular convolution: If $g = f \circledast h$ then

$$c_{k,l}(g) = c_{k,l}(f)\, c_{k,l}(h). \tag{1.19}$$

Shift:

$$f_{c,d}(x,y) \stackrel{\text{def}}{=} f(x+c, y+d) \Longleftrightarrow c_{k,l}(f_{c,d}) = e^{2\pi i(ck/X + ld/Y}\, c_{k,l}(f). \tag{1.20}$$

Differentiation:

$$c_{k,l}(f^{(s,t)}_{x,y}) = \left(\frac{2\pi i\, k}{X}\right)^s \left(\frac{2\pi i\, l}{Y}\right)^t c_{k,l}(f). \tag{1.21}$$

1.2 Periodic Discrete-Time Periodic Signals

The discrete-time counterparts of periodic signals are periodic sequences while the discrete Fourier transform (DFT) is the counterpart to the Fourier series. The fast Fourier transform (FFT) [1] algorithm is an important tool for signal processing.

1.2.1 One-Dimensional Periodic Discrete-Time Signals

The N-periodic real-valued sequences $\mathbf{x} \stackrel{\text{def}}{=} \{x[k]\},\ k \in \mathbb{Z},\ x[k+N] = x[k]$ are called discrete-time periodic signals. In the sequel, $N = 2^j$, j is a natural number and $\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$. These signals form an N-dimensional vector space, which is denoted by $\Pi[N]$, where the inner product and the norm are defined as

$$\langle \mathbf{x}, \mathbf{h} \rangle \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} x[k]\, h[k], \quad \|\mathbf{x}\| = \|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{k=0}^{N-1} |x[k]|^2}.$$

Note that any discrete-time signal $\mathbf{y}$ of limited duration L can be embedded into a space $\Pi[N],\ N \geq L$, by the periodization $\tilde{\mathbf{y}} \stackrel{\text{def}}{=} \left\{\tilde{y}[k] = \sum_{l \in \mathbb{Z}} y[k + lN]\right\}$.

The discrete circular convolution of two signals is an N-periodic signal

$$\mathbf{y} = \mathbf{h} \circledast \mathbf{x} \Longleftrightarrow y[k] = \sum_{l=0}^{N-1} h[k-l]\, x[l].$$

The discrete-time Fourier exponentials $\{\mathbf{e}_n\}$,

$$e_n[k] \stackrel{\text{def}}{=} e^{2\pi ink/N} = \omega^{kn}, n, k = 0, ..., N-1,$$

form an orthogonal basis of the space $\Pi[N]$:

$$\langle \mathbf{e}_n, \mathbf{e}_l \rangle = \sum_{k=0}^{N-1} e^{2\pi ink/N}\, e^{-2\pi ilk/N} = \sum_{k=0}^{N-1} e^{2\pi i(n-l)k/N} = \delta[n-l]\, N. \tag{1.22}$$

The symbol $\delta[k]$ denotes the Kronecker delta (impulse), that is the signal $\delta[k]$ is zero for all integers k except for $k = 0$ and $\delta[0] = 1$.

The expansion of a signal $\mathbf{x} = \{x[k]\}$ over the exponential basis

$$x[k] = \frac{1}{N} \sum_{n=0}^{N-1} \hat{x}[n]\, \omega^{nk} = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \hat{x}[n]\, \omega^{nk}, \quad \text{where } \hat{x}[n] = \sum_{k=0}^{N-1} x[k]\, \omega^{-nk}, \tag{1.23}$$

is the inverse discrete Fourier transform (IDFT) of the signal $\mathbf{x}$ and the set of coordinates $\left\{\hat{x}[n]\right\},\ n \in \mathbb{Z}$, is called the discrete Fourier transform (DFT) of $\mathbf{x}$. Both IDFT and DFT are computed by the fast Fourier transform (FFT) algorithm [1]. Since the signals are real-valued, the complex conjugate DFT is $\hat{x}[n]^* = \hat{x}[-n]$.

Throughout the book (*except for* Chap. 6), *the notation*

$$\hat{x}[n]_m = \sum_{k=0}^{N/2^m-1} x[k]\,\omega^{-2^m nk}, \quad x[k] = \frac{2^m}{N} \sum_{n=0}^{N/2^m-1} \hat{x}[n]_m\,\omega^{2^m nk}$$

is used for the DFT of signals belonging to the space $\Pi[N/2^m]$. *The sequence* $\left\{\hat{x}[n]_m\right\}$ *is* $N/2^m - periodic$.
The following relations between the signals from $\Pi[N]$ and their DFT hold:

Parseval identities: Equation (1.22) implies that

$$\langle \mathbf{x}, \mathbf{h} \rangle = \sum_{k=0}^{N-1} x[k]\,h[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \hat{x}[n]\,\hat{h}[-n],$$
$$\|\mathbf{x}\|^2 = \sum_{k=0}^{N-1} |x[k]|^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left|\hat{x}[n]\right|^2. \tag{1.24}$$

Circular discrete convolution: If $\mathbf{y} = \mathbf{x} \circledast \mathbf{h}$ then

$$\hat{y}[n] = \hat{h}[n]\,\hat{x}[n]. \tag{1.25}$$

Shift: If $\mathbf{x}_d \stackrel{\text{def}}{=} \{x[k+d]\}$, where d is an integer number, then $\hat{x}_d[n] = \omega^{nd}\,\hat{x}[n]$.
Finite differences The finite differences of a sequence $\mathbf{x}$ are defined iteratively: $\Delta[\mathbf{x}] \stackrel{\text{def}}{=} \{x[k+1] - x[k]\}$, $k \in \mathbb{Z}$, $\Delta^n[\mathbf{x}] \stackrel{\text{def}}{=} \Delta[\Delta^{n-1}[\mathbf{x}]]$. The central finite difference of second order $\delta^2[\mathbf{x}] \stackrel{\text{def}}{=} \{x[k+1] - 2x[k] + x[k-1]\}$, $k \in \mathbb{Z}$, and $\delta^{2r}[\mathbf{x}] \stackrel{\text{def}}{=} \delta^2[\delta^{2n-2}[\mathbf{x}]]$.

Proposition 1.1 *Assume that* $\mathbf{P}_{n-1} \stackrel{\text{def}}{=} \{P_{n-1}(k)\}$, $k \in \mathbb{Z}$, *is a sampled polynomial of degree not exceeding* $n-1$. *Then, for* $m < n$ $\Delta^m[\mathbf{P}_{n-1}]$ *is a sampled polynomial of degree not exceeding* $n - m - 1$. *In particular,* $\Delta^n[\mathbf{P}_{n-1}] = \mathbf{0}$. *If* n *is even then* $\delta^n[\mathbf{P}_{n-1}] = \mathbf{0}$.

Proof Denote $\mathbf{K}_l \stackrel{\text{def}}{=} \left\{k^l\right\}$, $k \in \mathbb{Z}$. The difference $\Delta[\mathbf{K}_l][k] = \left\{(k+1)^l - k^l\right\} = \left\{\sum_{m=0}^{l-1} \binom{l}{m} k^m\right\}$. Thus, application of the difference Δ reduces by one the degree of a sampled polynomial. Consequently, application of the difference Δ^{n-1} to a sampled polynomial $\mathbf{P_{n-1}}$ produces a constant, while application of Δ^n results in identical zero. ■

If $\mathbf{x}$ is a periodic discrete-time signal then its differences are periodic signals and their DFT coefficients are derived from the shift property of the DFT:

$$\widehat{\Delta^m[\mathbf{x}]}[n] = (\omega^n - 1)^m\,\hat{x}[n], \quad \widehat{\delta^{2l}[\mathbf{x}]}[n] = (-1)^l \left(2\sin\frac{\pi n}{N}\right)^{2l} \hat{x}[n]. \tag{1.26}$$

Polyphase representation of the DFT A signal $\mathbf{x} \stackrel{\text{def}}{=} \{x[k]\} \in \Pi[N],\ N = 2^j$, can be decomposed into a superposition of $M = 2^m,\ m < j$, signals belonging to $\Pi[N/M] = \Pi[N/2^m]$ by

$$\mathbf{x} = \bigcup_{l=0}^{M-1} \mathbf{x}_{l,M}, \quad x_{l,M}[k] \stackrel{\text{def}}{=} x[l + kM],\ k \in \mathbb{Z},$$

which is called the polyphase decomposition of the signal $\mathbf{x}$. The signals $\mathbf{x}_{l,M} \in \Pi[N/M]$ are the polyphase components of $\mathbf{x}$. The DFT of $\mathbf{x}$ can be split into the sum

$$\hat{x}[n] = \sum_{k=0}^{N-1} \omega^{-nk}\, x[k] = \sum_{m=0}^{M-1} \sum_{k=0}^{N/M-1} \omega^{-n(m+kM)}\, x[m + kM] = \sum_{m=0}^{M-1} \omega^{-ln}\, \widehat{x_{l,M}}[n]_m.$$

Special case: $M = 2$. Then, $\mathbf{x}_{0,2} = \{x[2k]\}$ and $\mathbf{x}_{1,2} = \{x[2k+1]\}$ are the even and odd components of the signal $\mathbf{x}$. The DFTs are, respectively,

$$\begin{aligned} &\hat{x}[n] = \hat{x}_{0,2}[n]_1 + \omega^{-n}\, \hat{x}_{1,2}[n]_1, \quad \hat{x}[n + N/2] = \hat{x}_{0,2}[n]_1 - \omega^{-n}\ \hat{x}_{1,2}[n]_1 \\ &\Longrightarrow \hat{x}_{0,2}[n]_1 = \frac{\hat{x}[n] + \hat{x}[n + N/2]}{2}, \quad \hat{x}_{1,2}[n]_1 = \frac{\hat{x}[n] - \hat{x}[n + N/2]}{2\omega^{-n}}. \end{aligned} \tag{1.27}$$

1.2.2 Two-Dimensional Periodic Discrete-Time Signals

Assume that j and l are natural numbers. Arrays $\mathbf{x} \stackrel{\text{def}}{=} \{x[k,n]\},\ k,n \in \mathbb{Z}$, which are periodic with the periods $K = 2^j$ and $N = 2^l$, $x[k + K, n + N] = x[k,n]$, are called two-dimensional periodic discrete-time signals. These signals form a KN-dimensional vector space, which is denoted by $\Pi[K,N]$, where the inner product and the norm are defined as

$$\langle \mathbf{x}, \mathbf{h} \rangle \stackrel{\text{def}}{=} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} x[k,n] h^*[k,n], \quad \|\mathbf{x}\| \stackrel{\text{def}}{=} \sqrt{\sum_{k=0}^{K-1} \sum_{n=0}^{N-1} |x[k,n]|^2}. \tag{1.28}$$

The discrete circular convolution of two signals from $\Pi[K,N]$ belongs to the same space:

$$\mathbf{y} = \mathbf{x} \circledast \mathbf{h} \Longleftrightarrow y[k,n] = \sum_{\kappa=0}^{K-1} \sum_{\nu=0}^{N-1} x[k-\kappa, n-\nu]\, h[\kappa, \nu]. \tag{1.29}$$

The two-dimensional discrete Fourier transform and its inverse are

$$\hat{x}[\kappa, \nu] = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} e^{-2\pi i(\kappa k/K + \nu n/N)} x[k, n] \tag{1.30}$$

$$x[k, n] = \frac{1}{KN} \sum_{\kappa=0}^{K-1} \sum_{\nu=0}^{N-1} e^{2\pi i(\kappa k/K + \nu n/N)} \hat{x}[\kappa, \nu]. \tag{1.31}$$

The following relations between the signals from $\Pi[K, N]$ and their DFT hold:

Parseval identities: Equation (1.22) implies that

$$\langle \mathbf{x}, \mathbf{h} \rangle = \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} x[k, n] h^*[k, n] = \frac{1}{KN} \sum_{\kappa=0}^{K-1} \sum_{\nu=0}^{N-1} \hat{x}[\kappa, \nu]\, \hat{h}^*[\kappa, \nu], \tag{1.32}$$

$$\|\mathbf{x}\|^2 = \frac{1}{KN} \sum_{\kappa=0}^{K-1} \sum_{\nu=0}^{N-1} \left|\hat{x}[\kappa, \nu]\right|^2. \tag{1.33}$$

Circular discrete convolution: If $\mathbf{y} = \mathbf{x} \circledast \mathbf{h}$ then $\hat{y}[\kappa, \nu] = \hat{h}[\kappa, \nu]\, \hat{x}[\kappa, \nu]$.

Shift: If $\mathbf{x}_{c,d} \stackrel{\text{def}}{=} \{x[k + c, n + d]\}$, $c, d \in \mathbb{Z}$, then $\widehat{x_{c,d}}[\kappa, \nu] = e^{2\pi i\,(\kappa c/K + \nu d/N)}\, \hat{x}[\kappa, \nu]$.

2D finite differences The finite differences $\Delta^{m,\mu}[\mathbf{x}]$ and $\delta^{m,\mu}[\mathbf{x}]$ for 2D periodic discrete-time signals $\mathbf{x}$ are defined as

$$\Delta^{1,1}[\mathbf{x}] \stackrel{\text{def}}{=} \Delta^1_{[k]}[\Delta^1_{[n]}[\mathbf{x}]] \Longleftrightarrow \Delta^{1,1}[\mathbf{x}][k, n] = x[k + 1, n + 1] - x[k + 1, n] - x[k, n + 1] + x[k, n], \quad \Delta^{m,\mu}[\mathbf{x}] \stackrel{\text{def}}{=} \Delta^m_{[k]}[\Delta^\mu_{[n]}[\mathbf{x}], \quad \delta^{m,\mu}[\mathbf{x}] \stackrel{\text{def}}{=} \delta^m_{[k]}[\delta^\mu_{[n]}[\mathbf{x}]. \tag{1.34}$$

They are 2D periodic signals and their DFT coefficients are:

$$\begin{aligned} \widehat{\Delta^{m,\mu}[\mathbf{x}]}[\kappa, \nu]] &= (e^{2\pi i\kappa/N} - 1)^m\, (e^{2\pi i\nu/N} - 1)^\mu\, \hat{\mathbf{x}}[\kappa.\nu], \\ \widehat{\delta^{m,\mu}[\mathbf{x}]}[\kappa, \nu] &= (-1)^{m+\mu} \left(2 \sin \frac{\pi\kappa}{N}\right)^m \left(2 \sin \frac{\pi\nu}{N}\right)^\mu \hat{\mathbf{x}}[\kappa, \nu]. \end{aligned} \tag{1.35}$$

References

1. J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex Fourier series. Math. Comput. **19**(90), 297–301 (1965)
2. A.V. Oppenheim, R.W. Schafer, *Discrete-Time Signal Processing*, 3rd edn. (Prentice Hall, New York, 2010)

Chapter 2
Introduction: Periodic Filters and Filter Banks

Abstract In this chapter filtering of periodic signals is outlined. Periodic filters and periodic filter banks are defined. Perfect reconstruction filter banks are characterized via their polyphase matrices.

2.1 Periodic Filters

Filtering of signals is a basic tool of signal processing. This procedure has some peculiarities in the periodic case, which are discussed in this section.

2.1.1 Definition of Periodic Filters

A linear operator $\mathbf{h}{:}\Pi[N] \longrightarrow \Pi[N]$, $\mathbf{y} = \mathbf{hx}$, is time-invariant if the integer shift $\mathbf{x}_d \stackrel{\text{def}}{=} \{x[k+d]\}$ of the input signal produces the same shift $\mathbf{y}_d \stackrel{\text{def}}{=} \{y[k+d]\}$ of the output. Such operators are called digital periodic filters (p-filters).

Denote by $\tilde{\delta}[k] \stackrel{\text{def}}{=} \delta[k](\mathrm{mod}N)$ the N-periodized Kronecker delta. If the input signal $\mathbf{i} = \left\{\tilde{\delta}[k]\right\}$, then the output signal $\mathbf{hi} = \{h[k]\}$, which, obviously, belongs to $\Pi[N]$, is called the *impulse response* (IR) of the p-filter $\mathbf{h}$. In the sequel, we use the notation $\mathbf{h}$ for both the p-filter and its IR $\{h[k]\}$. If the IR of a p-filter is finite up to periodization then the p-filter is referred to as the periodic finite impulse response p-filter (p-FIR p-filters). Otherwise, the p-filter is referred to as the infinite impulse response p-filter (IIR p-filters).

Any periodic signal $\mathbf{x}$ can be represented as a linear combination of impulses $x[k] = \sum_{l=0}^{N-1} x[l]\,\tilde{\delta}[k-l] = \mathbf{x} \circledast \mathbf{i}$. Then, due to the time-invariance of the filter $\mathbf{h}$, the output signal becomes $\mathbf{y} = \mathbf{hx} = \mathbf{x} \circledast \mathbf{h} \Longleftrightarrow y[k] = \sum_{l=0}^{N-1} h[k-l]\,x[l]$. Thus, application of the p-filter to a periodic signal is implemented via the discrete circular convolution. The DFT of the IR of the p-filter $\mathbf{h}$ is called its *frequency response* (FR)

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_2,

$$\hat{\mathbf{h}} = \left\{\hat{h}[n]\right\}, \quad \hat{h}[n] = \sum_{k=0}^{N-1} h[k]\,\omega^{-nk}, \quad n \in \mathbb{Z}.$$

Periodic filtering of a signal reduces to multiplication in the frequency domain: $\mathbf{h} \circledast \mathbf{x} \Longleftrightarrow \hat{h}[n]\,\hat{x}[n], n \in \mathbb{Z}$. The frequency response of a filter can be represented in a polar form $\hat{h}[n] = \left|\hat{h}[n]\right| e^{i \arg(\hat{h}[n])}$, where the positive N-periodic sequence $\left|\hat{h}[n]\right|$ is called the *magnitude response* (MR) of $\mathbf{h}$ and the real-valued 2π-periodic sequence $\arg(\hat{h}[n])$ is called the *phase response* of $\mathbf{h}$. A p-filter is referred to as a linear phase one if its phase response is linear in n. If the IR of a p-filter $\mathbf{h}$ is symmetric or antisymmetric within the interval $k = -N/2, \ldots, N/2 - 1$ then $\mathbf{h}$ is a linear phase p-filter.

A p-filter $\mathbf{h}$ is called low-pass if it passes low frequencies of signals but attenuates or completely rejects frequencies higher than a cutoff frequency. Its FR $\hat{h}[0] \neq 0$, $\hat{h}[\pm N/2]$ is close or equal to zero. On the contrary, a high-pass p-filter $\mathbf{g}$ attenuates or rejects frequencies lower than the cutoff frequency. Its frequency response $\hat{g}[\pm N/2] \neq 0$, while $\hat{h}[0]$ is close or equal to zero. A band-pass filter passes frequencies within a certain range (passband) and attenuates (rejects) frequencies outside that range (stopband).

2.1.2 Multirate p-Filtering

Assume that $\mathbf{x} \stackrel{\text{def}}{=} \{x[k]\}$ belongs to $\Pi[N]$, $N = 2^j$, and $M = 2^m$, $m < j$. The operation $(\downarrow M)\mathbf{x} = \grave{\mathbf{x}} \stackrel{\text{def}}{=} \{x[Mk]\} \in \Pi[N/M]$ is called downsampling the signal $\mathbf{x}$ by factor of M. Assume that a signal $\mathbf{x}$ belongs to $\Pi[N/M]$. The operation

$$(\uparrow M)\mathbf{x} = \acute{\mathbf{x}}, \quad \acute{x}[k] = \begin{cases} x[l], & \text{if } k = lM; \\ 0, & \text{otherwise.} \end{cases}, \quad l \in \mathbb{Z},$$

is called upsampling the signal $\mathbf{x}$ by factor of M. Obviously, the downsampled signal $(\downarrow M)\mathbf{x} = \mathbf{x}_{0,M}$ is the zero polyphase component of the signal $\mathbf{x}$, while the upsampled signal $(\uparrow M)\mathbf{x}$ is a signal, whose all the polyphase components, except for the zero component, vanish.

If p-filtering a signal is accompanied by downsampling or upsampling then it is called multirate p-filtering. Let $\mathbf{h} = \{h[k]\}$ and $\tilde{\mathbf{h}} = \left\{\tilde{h}[k]\right\}$ be p-filters. Application of the p-filter $\bar{\mathbf{h}} = \left\{\tilde{h}[-k]\right\}$, which is the time-reversed p-filter $\tilde{\mathbf{h}}$, to a signal $\mathbf{x}$, which is followed by downsampling by factor of M, produces the signal $\mathbf{y} = \{y[k]\} \in \Pi[N/M]$, $y[k] = \sum_{l=0}^{N-1} \tilde{h}[l - Mk]\,x[l]$, which is the zero polyphase component of the signal $\mathbf{y} = \bar{\mathbf{h}} \circledast \mathbf{x}$. Application of the p-filter $\mathbf{h}$ to a signal $\check{\mathbf{y}} = \{\check{y}[k]\} \in \Pi[N/M]$, which is upsampled by factor of M, produces the signal

$$\check{\mathbf{x}} = \{\check{x}[k]\} \in \Pi[N], \quad \check{x}[k] = \sum_{l=0}^{N/M-1} h[k - Ml]\, \check{y}[l] \Longleftrightarrow \hat{\check{x}}[n] = \hat{h}[n]\, \hat{\check{y}}[Mn].$$

In the rest of the volume we mainly deal with down(up)sampling by factor $M = 2$ and denote by $\mathbf{x}_0 = \mathbf{x}_{0,2}$ and $\mathbf{x}_1 = \mathbf{x}_{1,2}$ the even and the odd polyphase components of a signal $\mathbf{x}$.

$M = 2$: **Downsampling:** The downsampled signal $\check{\mathbf{y}} = \mathbf{y}_0$ is the even polyphase component of the signal $\mathbf{y} = \{y[k]\}$, $y[k] = \sum_{l=0}^{N-1} \tilde{h}[l-k]\, x[l]$, whose DFT is $\hat{y}[n] = \hat{\tilde{h}}[-n]\, \hat{x}[n]$. Using Eq. (1.27), we get

$$\begin{aligned}\hat{\tilde{h}}[-n]\, \hat{x}[n] &= \hat{\tilde{h}}_0[-n]_1\, \hat{x}_0[n]_1 + \hat{\tilde{h}}_1[-n]_1\, \hat{x}_1[n]_1 \\ &\quad + \omega^{-n} \left(\widehat{\tilde{h}_0}[-n]_1\, \hat{x}_1[n]_1 + \hat{\tilde{h}}_1[-n]_1\, \hat{x}_0[2(n+1)]\right) \\ \Longrightarrow \hat{y}[n] &= \hat{\tilde{h}}_0[-n]_1\, \hat{x}_0[n]_1 + \hat{\tilde{h}}_1[-n]_1\, \hat{x}_1[n]_1, \quad n \in \mathbb{Z}. \end{aligned} \tag{2.1}$$

$M = 2$: **Upsampling:** Assume that p-filter $\mathbf{h}$ is applied to a signal $\check{\mathbf{y}} = \{\check{y}[k]\} \in \Pi[N/2]$, which is upsampled by factor of 2. Then,

$$\begin{aligned}\check{x}[k] = \sum_{l=0}^{N/2-1} h[k-2l]\, \check{y}[l] \Longleftrightarrow \hat{\check{x}}[n] &= \hat{h}[n]\, \hat{\check{y}}[n]_1 = \left(\hat{h}_0[n]_1 + \omega^{-n} \hat{h}_1[n]_1\right)\, \hat{\check{y}}[n]_1 \\ \Longrightarrow \hat{\check{x}}_0[n]_1 &= \hat{h}_0[n]_1\, \hat{\check{y}}[n]_1, \quad \hat{\check{x}}_1[n]_1 = \hat{h}_1[n]_1\, \hat{\check{y}}[n]_1. \end{aligned} \tag{2.2}$$

Interpolating p-filters If the DFT of the even polyphase component of a p-filter $\mathbf{h}$ is constant $\hat{h}_0[n]_1 \equiv C$ then the p-filter is called interpolating. In that case, Eq. 2.2 implies that the DFT of the zero polyphase component of the output signal $\check{\mathbf{x}}$ is $\hat{\check{x}}_0[n]_1 = C\, \hat{\check{y}}[n]_1$. This means that $\check{x}[2k] = C\, \check{y}[k],\ k \in \mathbb{Z}$.
Example **Butterworth p-filters**: Denote

$$f^{2r}[n]_1 \stackrel{\text{def}}{=} \frac{\omega^n \left(\cos^{2r} \pi n/N - \sin^{2r} \pi n/N\right)}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \tag{2.3}$$

Obviously the denominator of the $N/2$-periodic sequence $f^{2r}[n]_1$ is strictly positive and $f^{2r}[0]_1 = 1$. Therefore, the sequences

$$\hat{h}[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(1 + \omega^{-n} f_d^{2r}[n]_1\right) = \frac{\sqrt{2} \cos^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}, \tag{2.4}$$

$$\hat{g}[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(1 - \omega^{-n} f_d^{2r}[n]_1\right) = \frac{\sqrt{2} \sin^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N} \tag{2.5}$$

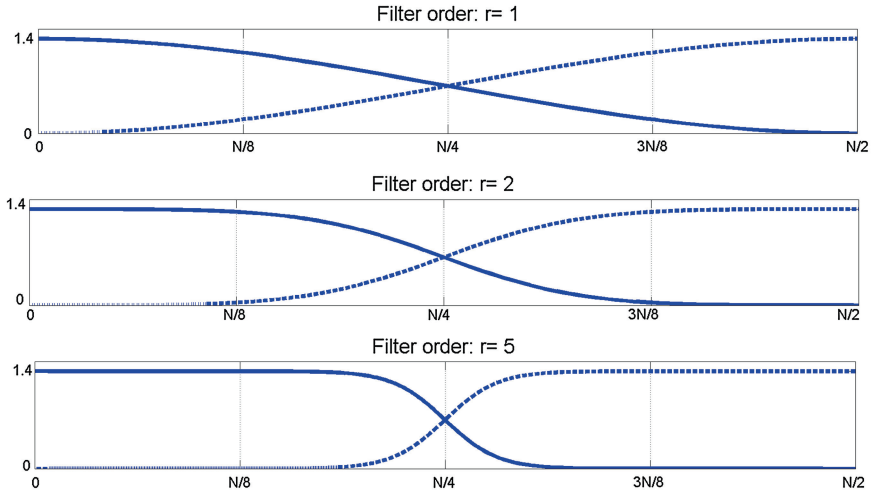


Fig. 2.1 The FR of the low-pass p-filters $\mathbf{h}$ (*solid lines*) and of the high-pass p-filters $\mathbf{g}$ (*dashed lines*) for orders $r = 1, 2, 5$

can serve as the FR of interpolating p-filters $\mathbf{h}$ and $\mathbf{g}$. The FR $\hat{h}[0] = \sqrt{2}, /; \hat{h}[N/2] = 0$. Therefore, the p-filter $\mathbf{h}$ is low-pass. On the contrary, $\hat{g}[0] = 0, /; \hat{g}[N/2] = \sqrt{2}$. Therefore, the p-filter $\mathbf{g}$ is high-pass.

The sequences $\hat{h}[n]$ and $\hat{g}[n]$ are the magnitude squared FR of the periodized half-band low- and high-pass Butterworth filters of order r, respectively. The Butterworth filters are widely used in signal processing [1]. Figure 2.1 displays the magnitude squared FR of the half-band low- and high-pass Butterworth filters of orders $r = 1, 2, 5$. One can observe that these frequency responses, especially of the order 5 filters, are flat within their pass-band, while practically vanishing at the stop-band.

2.2 Periodic Filter Banks

Keeping in mind the forthcoming periodic wavelet and wavelet frame transforms, we introduce periodic filter banks.

2.2.1 Filter Banks

Definition 2.1 The set of p-filters $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s\right\}$, $s = 0, \ldots, S-1$, which, being time-reversed and applied to an input signal $\mathbf{x} \in \Pi[N]$, produces the set of the output signals $\{\tilde{\mathbf{y}}^s\}_{s=0}^{S-1}$, which are downsampled by factor of M,

$$y^s[l] = \sum_{k=0}^{N-1} \tilde{h}^s[k - Ml]\, x[k], \quad s = 0, \ldots, S-1,\ l \in \mathbb{Z}, \tag{2.6}$$

is called the $S-$channel analysis p-filter bank.

Definition 2.2 The set of p-filters $\mathbf{H} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s\right\},\ s = 0, \ldots, S-1$, which, being applied to a set of input signals $\{\mathbf{y}^s\} \in \Pi[N/M],\ s = 0, \ldots, S-1$, that are upsampled by factor of M, produces the output signal $\check{x}[l]) = \sum_{s=0}^{S-1} \sum_{k=0}^{N/M-1} h^s[l - Mk]\ y^s[k],\ \ l \in \mathbb{Z}$, is called the $S-$channel synthesis p-filter bank.

Definition 2.3 If the upsampled signals $\mathbf{y^s},\ s = 0, \ldots, S-1$, which are defined in Eq. 2.6, are used as an input to the synthesis p-filter bank and the output signal is $\check{\mathbf{x}} = \mathbf{x}$, then the pair of analysis–synthesis p-filter banks form a perfect reconstruction (PR) p-filter bank.

Definition 2.4 If the number of channels S equals to the downsampling factor M then the p-filter bank is said to be critically sampled. If $S > M$ then the p-filter bank is oversampled.

Critically sampled PR p-filter banks are used in the wavelet analysis, while oversampled PR p-filter banks serve as a source for discrete-time wavelet frames design.

In the volume, we deal mainly with p-filter banks, whose downsampling factor is $M = 2$ *and* $\tilde{\mathbf{h}}^0$ *and* $\mathbf{h}^0$ *are the low-pass p-filters.*

2.2.2 Characterization of p-Filter Banks

Assume that $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s\right\}, s = 0, \ldots, S-1$, is an analysis p-filter bank with the downsampling factor of 2. Then, its application to a signal $\mathbf{x} \in \Pi[N]$ produces S signals from $\Pi[N/2]$:

$$y^s[l] = \sum_{k=0}^{N-1} \tilde{h}^s[k - 2l]\, x[k], \quad s = 0, \ldots, S-1,\ l \in \mathbb{Z}. \tag{2.7}$$

It follows from Eq. (2.1) that

$$\hat{y}^s[n]_1 = \hat{\tilde{h}}_0^s[-n]_1\, \hat{x}_0[n]_1 + \hat{\tilde{h}}_1^s[-n]_1\, \hat{x}_1[n]_1, \quad n \in \mathbb{Z}. \tag{2.8}$$

Assume that $\mathbf{H} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s\right\},\ s = 0, \ldots, S-1$, is a synthesis p-filter bank with the upsampling factor of 2. Then, its application to the upsampled signals $\mathbf{y}^s \in \Pi[N/2]$ produces a signal from $\Pi[N]$:

$$\check{x}[l]) = \sum_{s=0}^{S-1} \sum_{k=0}^{N/2-1} h^s[l - 2k]\ y^s[k], \quad l \in \mathbb{Z}. \tag{2.9}$$

Equation (2.2) implies that the DFTs of the polyphase components are

$$\hat{\tilde{x}}_0[n]_1 = \sum_{s=0}^{S-1} \hat{h}_0^s[n]_1\, \hat{y}^s[n]_1, \quad \hat{\tilde{x}}_1[n]_1 = \sum_{s=0}^{S-1} \hat{h}_1^s[n]_1\, \hat{y}^s[n]_1. \tag{2.10}$$

Equations (2.8) and (2.10) can be represented in a matrix form by

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix} = \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}, \quad \begin{pmatrix} \hat{\tilde{x}}_0[n]_1 \\ \hat{\tilde{x}}_1[n]_1 \end{pmatrix} = \mathbf{P}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix}$$

where the $S \times 2$ analysis and the $2 \times S$ synthesis polyphase matrices are, respectively,

$$\tilde{\mathbf{P}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^0[n]_1 & \hat{\tilde{h}}_1^0[n]_1 \\ \vdots & \vdots \\ \hat{\tilde{h}}_0^{S-1}[n]_1 & \hat{\tilde{h}}_1^{S-1}[n]_1 \end{pmatrix} \quad \mathbf{P}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 \ \ldots \ \hat{h}_0^{S-1}[n]_1 \\ \hat{h}_1^0[n]_1 \ \ldots \ \hat{h}_1^{S-1}[n]_1 \end{pmatrix}, \quad n \in \mathbb{Z}.$$

If the relations

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \mathbf{I}_2, \tag{2.11}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix, holds for all $n \in \mathbb{Z}$ then

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix} = \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}.$$

Thus, Eq. (2.11) is the condition for the analysis–synthesis pair $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$ of p-filter banks to form a PR p-filter bank.

Reference

1. A.V. Oppenheim, R.W. Schafer, *Discrete-Time Signal Processing*, 3rd edn. (Prentice Hall, New York, 2010)

Chapter 3
Mixed Circular Convolutions and Zak Transforms

Abstract In this chapter the notion of mixed circular convolution is introduced. The polynomial and discrete periodic splines defined on uniform grids are special cases of such convolutions. The so-called Zak transforms provide tools to handle mixed circular convolutions

3.1 Periodic Discrete-Continuous Circular Convolution and Zak Transform

Assume that $f(t)$ is a continuous-time N-periodic signal and $\mathbf{q} \stackrel{\text{def}}{=} \{q[k]\}$ is a discrete-time N-periodic signal, where N is a natural number. The function

$$g(t) = \mathbf{q} \circledast f(t) \stackrel{\text{def}}{=} \sum_{l=0}^{N-1} q[l] f\,(t-l) \tag{3.1}$$

is referred to as the periodic mixed discrete-continuous convolution. Obviously, the function $g(t)$ is a continuous-time N-periodic signal.

Assume that Fourier coefficients $c_n(f)$ of the periodic signal $f(t)$ decay sufficiently fast for the subsequent formulas to be justifiable. For example, it is sufficient for $f(t)$ to a have piece-wise continuous derivative. Due to Eqs. (1.3) and (1.5),

$$\begin{aligned} f\,(t-l) &= \frac{1}{N}\sum_{n\in\mathbb{Z}} c_n(f)\, e^{2\pi i\, n(t-l)/N} = \frac{1}{N}\sum_{m\in\mathbb{Z}}\sum_{n=-N/2}^{N/2-1} c_{n+mN}(f)\, e^{2\pi i(n+Nm)(t-l)/N} \\ &= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} Z[f]_n(t)\, e^{-2\pi i n l/N}, \quad f\,(t+l) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} Z[f]_n(t)\, e^{2\pi i n l/N}, \end{aligned} \tag{3.2}$$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_3,

where

$$Z[f]_n(t) \stackrel{\text{def}}{=} \sum_{m\in\mathbb{Z}} c_{n+mN}(f)\, e^{2\pi i(n+Nm)t/N}, \quad n = -N/2, ..., N/2-1. \tag{3.3}$$

We derive a couple of observations from Eqs. (3.2) and (3.3).

1. For any fixed value t, the sequence $Z[f]_n(t)$ is N-periodic with respect to the variable n.
2. For any fixed value n, the function $Z[f]_n(t)$ is N-periodic with respect to the variable t.
3. Let $f_d(t) \stackrel{\text{def}}{=} f(t+d),\ d \in \mathbb{Z}$, be a circular shift of the signal f. Then,

$$Z[f_d]_n(t) = e^{2\pi i n d/N} Z[f]_n(t). \tag{3.4}$$

4. Equation (3.3) provides the Fourier expansion of the function $Z[f]_n(t)$. Thus, the Fourier coefficients

$$c_m\left(Z[f]_n\right) = N \begin{cases} c_m(f), & \text{if } m = n + lN, \\ 0, & \text{otherwise,} \end{cases} = N\delta[m-n](\text{mod } N)\, c_m(f). \tag{3.5}$$

5. On the other hand, Eq. (3.2) implies that, by a fixed t, $f(t+k)$ is the IDFT of $Z[f]_n(t)$ and, consequently, $Z[f]_n(t)$ is the DFT of $f(t+k)$:

$$Z[f]_n(t) = \sum_{k=0}^{N-1} f(t+k)\, e^{-2\pi i n k/N} = \sum_{k=0}^{N-1} f(t-k)\, e^{2\pi i n k/N}, \tag{3.6}$$

where $n = -N/2, ..., N/2-1$.

Definition 3.1 The set of N-periodic functions $Z[f]_n(t),\ n = -N/2, ..., N/2-1$, which is defined by Eq. (3.3) (or, by equivalent Eq. (3.6)) is referred to as the periodic Zak transform of an N-periodic signal $f(t)$ [2, 15].

Combining different representations (Eqs. 3.3 and 3.6) of the periodic Zak transform, we get the relations for $n = -N/2, ..., N/2-1$:

$$Z[f]_n(t) = \sum_{k=0}^{N-1} f\ (t+k)\ e^{-2\pi i n k/N} = \sum_{l\in\mathbb{Z}} c_{n+lN}(f)\, e^{2\pi i(n+Nl)t/N}. \tag{3.7}$$

It implies the following special cases:

$$Z[f]_n(0) = \sum_{k=0}^{N-1} f\ (k)\ e^{-2\pi i n k/N} = \sum_{l\in\mathbb{Z}} c_{n+lN}(f), \tag{3.8}$$

$$Z[f]_0(t) = \sum_{k=0}^{N-1} f\,(t+k) = \sum_{l\in\mathbb{Z}} c_{lN}(f)\, e^{2\pi i l t}. \tag{3.9}$$

Equation (3.8) establishes relation between the Fourier coefficients of a periodic function and the DFT of its sampling. Equation (3.9) can be regarded as a periodic counterpart of the Poisson summation formula ([6, 10], for example).

Let $f(t)$ be an N-periodic signal and T be a positive number. Then the signal $f(Nt/T)$ is T-periodic (belongs to $\mathscr{P}_T$). Denote by $\mathscr{M}_T[f]$ the subspace of the space $\mathscr{P}_T$, which consists of mixed convolutions of different discrete-time signals $\mathbf{q} \in \Pi[N]$ with the signal $f(Nt/T) \in \mathscr{P}_T$:

$$g \in \mathscr{M}_T[f] \Longleftrightarrow g(t) = \sum_{l=0}^{N-1} q[l] f\left(\frac{Nt}{T} - l\right), \quad \text{where } \mathbf{q} \in \Pi[N]. \tag{3.10}$$

The function $f \in \mathscr{P}_N$ is called the generator of $\mathscr{M}_T[f]$.

Remark 3.1.1 If a signal $g(t)$ belongs to the space $\mathscr{M}_T[f]$ then the same is true for all its "$h\times$integer", $h = T/N$, shifts $g(t + hd)$, $d \in \mathbb{Z}$. Thus, $\mathscr{M}_T[f]$ is a *shift-invariant space* [7].

The set $\{Z[f]_n(t)\}$ provides a proper framework for the interpolation.

Proposition 3.1 *Assume the sequence* $\left\{Z[f]_n(0) = \sum_{l=0}^{N-1} f(k)\, e^{-2\pi i n l/N}\right\}$ *does not vanish as* $n = -N/2, ..., N/2-1$. *Then, any signal* $\mathbf{x} \in \Pi[N]$ *can be uniquely interpolated by an element* $g(t) \in \mathscr{M}_T[f]$, *that is* $g(Tk/N) = x[k]$, $k = 0, ..., N-1$.

Proof Let $g(t) = \sum_{l=0}^{N-1} q[l] f(Nt/T - l)$. Then, $g(Tk/N) = \sum_{l=0}^{N-1} q[l] f(k-l)$ is the discrete circular convolution of the signal $\mathbf{q} \in \Pi[N]$ with the sampled signal $f(t)$. Applying the DFT to both parts of the relation $g(Tk/N) = x[k]$, $k = 0, ..., N-1$, we get, due to Eq. (1.25),

$$\hat{q}[n] \sum_{l=0}^{N-1} f(l)\, e^{-2\pi i n l/N} = \hat{x}[n] \Longrightarrow \hat{q}[n] = \frac{\hat{x}[n]}{Z[f]_n(0)}, \tag{3.11}$$

$$q[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \hat{q}[n]\, e^{2\pi i n k/N}.$$

■

Corollary 3.1 *If the sequence* $\{Z[f]_n(0)\}$, $n = -N/2, ..., N/2-1$, *does not vanish then the space* $\mathscr{M}_N[f]$ *is* N*-dimensional. Consequently, the functions* $\{f(Nt/T - l)\}$, $l = 0, ..., N-1$ *are linearly independent and form a basis of the space* $\mathscr{M}_T[f]$. *In addition, the signals*

$$\boldsymbol{f}_l = \{f_l[k]\},\ f_l[k] \stackrel{\text{def}}{=} \{f(k-l)\},\ k = 0, ..., N-1,\ l \in \mathbb{Z},$$

form a basis of the discrete-time signals space $\Pi[N]$.

The sequence $\{Z[f]_n(0)\},\ n = -N/2, ..., N/2-1$ is called the characteristic sequence of the space $\mathscr{M}_N[f]$.

Proposition 3.2 *The signals* $\left\{Z_T[f]_n(t) \stackrel{\text{def}}{=} Z[f]_n(Nt/T)\right\} \in \mathscr{P}_T, l = 0, ..., N-1$, *form an orthogonal basis of the space* $\mathscr{M}_T[f]$.

Proof Equations (1.8) and (3.5) imply that

$$\begin{aligned}\langle Z_T[f]_n, Z_T[f]_r\rangle &= \int_0^T Z_T[f]_n(t)\, Z_T^*[f]_r(t)\,\mathrm{d}t = \frac{T}{N}\int_0^N Z[f]_n(t)\, Z^*[f]_r(t)\,dt \\ &= \frac{T}{N^2}\sum_{m\in\mathbb{Z}} c_m(Z[f]_n)\, c_m^*(Z[f]_r) \\ &= T\delta[n-r]\sum_{l\in\mathbb{Z}} |c_{n+lN}(f)|^2.\end{aligned}$$

Assume $g(t) = \sum_{l=0}^{N-1} q[l] f(Nt/T - l) \in \mathscr{M}_T[f]$. Equation (3.2) implies that

$$g(t) = \frac{1}{N}\sum_{l=0}^{N-1} q[l] \sum_{n=-N/2}^{N/2-1} e^{-2\pi i n l/N}\, Z_T[f]_n(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi_n\, Z_T[f]_n(t),$$

$$\xi_n = \hat{q}[n] = \sum_{l=0}^{N-1} q[l]\, e^{-2\pi i n l/N} \Longrightarrow q[l] = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi_n\, e^{2\pi i n l/N}.$$

Thus, coordinates of different representations of a signal belonging to the space $\mathscr{M}_T[f]$ are linked to each other via the direct and inverse DFT. ■

3.2 A Leading Example: Periodic Polynomial Splines

Periodic polynomial splines provide an example of mixed discrete-continuous circular convolution.

Definition 3.2 An N-periodic function $S^p(t)$ is called the periodic spline of order $p \in \mathbb{N}$ on the grid $\{k\},\ k \in \mathbb{Z}$, if it has $p-2$ continuous derivatives (belongs to C^{p-2}) and consists of pieces of polynomials of degree $p-1$ that are linked to each other at the nodes. Nodes of splines of even and of odd orders are located at points $\{k\}$ and $\{(k+1)/2\},\ k \in \mathbb{Z}$, respectively.

The spline spaces may have a variety of generating functions, best known of whose are the B-splines.

3.2.1 Periodic B-Splines

Let $N = 2^j,\ j \in \mathbb{N}$. The centered N-periodic B-spline $B^1(t)$ of first order on the grid $\{k\},\ k \in \mathbb{Z}$, is defined via periodization of the compactly supported function $\chi[-1/2, 1/2](t)$, $B^1(t) \stackrel{\text{def}}{=} \sum_{l\in\mathbb{Z}} \chi[-1/2, 1/2](t + Nl)$, where

$$\chi[a, b](t) \stackrel{\text{def}}{=} \begin{cases} 1, \text{ if } t \in (a, b); \\ 0, \text{ otherwise,} \end{cases} \tag{3.12}$$

is the so-called characteristic function of the interval (a, b).

The Fourier coefficients of the B-spline

$$c_n(B^1) = \int_{-N/2}^{N/2} B^1(t)\, e^{-2\pi int/N} = \frac{1}{2}\int_{-1/2}^{1/2} e^{-2\pi int/N} = \frac{\sin \pi n/N}{\pi n/N}.$$

The B-splines of higher order are defined iteratively via the circular convolution $B^p(t) \stackrel{\text{def}}{=} B^1 \circledast B^{p-1}(t)$. It follows from Eq. (1.9) that the B-spline $B^p(t)$ can be expanded into the Fourier series

$$c_n(B^p) = \left(\frac{\sin \pi n/N}{\pi n/N}\right)^p \Longrightarrow B^p(t) = \frac{1}{N}\sum_{n\in\mathbb{Z}} e^{2\pi int/N}\left(\frac{\sin \pi n/N}{\pi n/N}\right)^p. \tag{3.13}$$

The B-spline $B^p(t)$ is supported on the interval $(-p/2,\ p/2)$ (up to periodization). It is strictly positive inside this interval and symmetric about zero, where it has its single maximum. The B-spline B^p consists of pieces of polynomials of degree $p-1$ that are linked to each other at the nodes such that $B^p \in C^{p-2}$. Nodes of B-splines of even and of odd orders are located at points $\{k\}$ and $\{(k+1)/2\},\ k \in \mathbb{Z}$, respectively. There exists an explicit expression for the B-spline at the interval $(-N/2, N/2)$:

$$B^p(t) = \frac{1}{(p-1)!}\sum_{k=0}^{p}(-1)^p \binom{p}{k}\left(t + \frac{p}{2} - k\right)_+^{p-1}, \quad t_+ \stackrel{\text{def}}{=} \begin{cases} t, \text{ if } t \geq 0; \\ 0, \text{ otherwise.} \end{cases} \tag{3.14}$$

Surely, the length $p+1$ of the spline's B^p support should not exceed the period N, thus $p+1 < N$. Figures 3.1 and 3.2 display the B-splines of different orders and their Fourier spectra.

Remark 3.2.1 Visually, the B-spline curves tend to Gaussian as the spline order increases. It was proved in [12] that

$$\lim_{p\to+\infty}\left\{\sqrt{\frac{p}{12}}\, B^p\left(\frac{p}{12}t\right)\right\} = \frac{1}{\sqrt{2\pi}}\exp(-t^2/2).$$

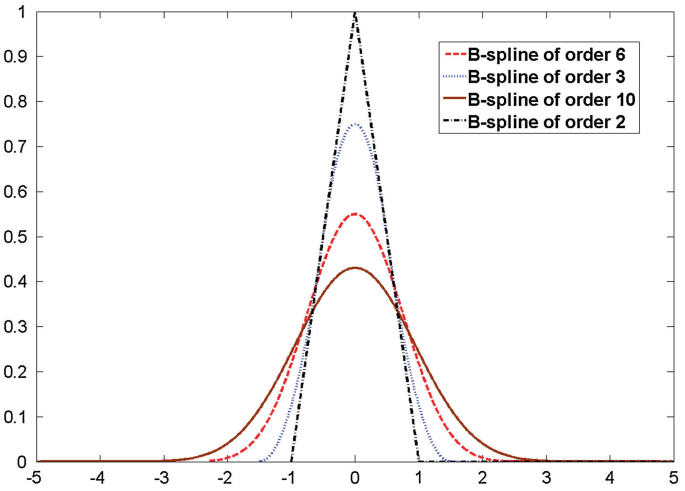


Fig. 3.1 B-splines of orders 2 (piecewise linear), 3, 6 and 10

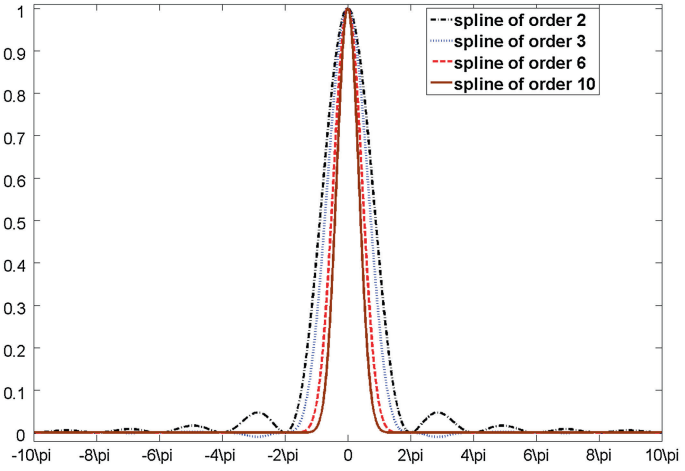


Fig. 3.2 Fourier spectra of the B-splines of orders 2, 3, 6 and 10

Figure 3.3 displays the B-spline $B_1^{12}(t)$ of twelfth order and the Gaussian curve $(2\pi)^{-1/2}\exp(-t^2/2)$. We observe that the curves are hardly distinguishable.

Direct calculation of the B-splines values using Eq. (3.14) is implemented by MATLAB function `bsplP.m`. Figures 3.1 and 3.2 were produced by the MATLAB codes `bsplsP.m` and `bsplFP.m`, respectively.

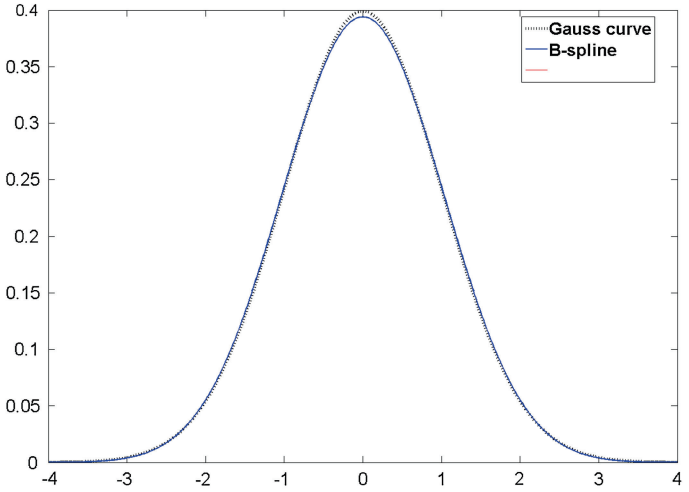


Fig. 3.3 *Solid line*: B-splines of order 12. *Dotted line*: Gauss curve

3.2.2 Spaces of Periodic Splines

Let T be a positive number. The periodic B-splines generate the shift-invariant T-periodic splines spaces ${}^{p}\mathscr{S}_T$:

$$S_T^p(t) \in {}^{p}\mathscr{S}_T \Longleftrightarrow S^p(t) = \mathbf{q} \circledast B^p\left(\frac{Nt}{T}\right) = \sum_{k=0}^{N-1} q[k]\, B^p\left(\frac{Nt}{T} - k\right), \quad (3.15)$$

where $\mathbf{q}$ is an N-periodic discrete- time signal. The functions $S_T^p(t) \in {}^{p}\mathscr{S}_T$ are T-periodic splines of order p, whose nodes are located at points $\{Tk/N\}$ when p is even and at points $\{T(k+1/2)/N\},\ k \in \mathbb{Z}$ when p is odd.

The Fourier coefficients of the spline $S_T^p(t)$ are

$$\begin{aligned} c_n[S_T^p] &= \int_0^T e^{-2\pi nt/T} \sum_{k=0}^{N-1} q[k]\, B^p\left(\frac{Nt}{T} - k\right) \mathrm{d}t \qquad (3.16)\\ &= \frac{T}{N} \sum_{k=0}^{N-1} e^{-2\pi nk/N}\, q[k] \int_0^N e^{-2\pi nt/N}\, B^p(t)\, \mathrm{d}t \\ &= \frac{T}{N}\, \hat{q}[n] \left(\frac{\sin \pi\,(n/N)}{\pi\,(n/N}\right)^p. \end{aligned}$$

Due to Eq. (3.7), the Zak transform of the periodic B-spline is

$$\zeta^p[n](t) \stackrel{\text{def}}{=} Z[B^p]_n(t) = \sum_{k=0}^{N-1} e^{2\pi i n k/N} B^p(t-k) \tag{3.17}$$
$$= \sum_{l\in\mathbb{Z}} e^{2\pi i (n+Nl)t/N} \left(\frac{\sin\pi(n/N+l)}{\pi(n/N+l)}\right)^p.$$

The functions $\{\zeta^p[n](Nt/T)\}$, $n = -N/2, ..., N/2-1$ are the splines from the space ${}^p\mathscr{S}_T$. We refer to the splines $\zeta^p[n](t)$ as the *periodic exponential splines*.

Due to Eq. (3.2),

$$B^p(t-k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \zeta^p[n](t)\, e^{-2\pi i n k/N}, \quad B^p(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \zeta^p[n](t). \tag{3.18}$$

Proposition 3.3 *The following relations for the B-splines of any order p hold*

$$\int_0^N B^p(\tau)\, d\tau = \sum_{k=0}^{N-1} B^p(t-k) \equiv 1. \tag{3.19}$$

Proof Both relations in Eq. (3.19) follow from the expression Eq. (3.13) for the Fourier coefficients of the B-spline. The integral

$$\int_0^N B^p(\tau)\, d\tau = c_0(B^p) = 1.$$

In turn, Eq. (3.9) implies that the sum

$$\sum_{k=0}^{N-1} B^p(t-k) = \sum_{l\in\mathbb{Z}} c_{lN}(B^p)\, e^{-2\pi i l t} = c_0(B^p) = 1.$$

■

The characteristic sequence of the space ${}^p\mathscr{S}_T$

$$u^p[n] \stackrel{\text{def}}{=} \zeta^p[n](0) = \sum_{k=0}^{N-1} e^{-2\pi i n k/N} B^p(k) \tag{3.20}$$
$$= \sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi(n/N+l)}{\pi(n/N+l)}\right)^p = \sin^p\frac{\pi n}{N} \sum_{l\in\mathbb{Z}} \left(\frac{(-1)^l}{\pi(n/N+l)}\right)^p.$$

Equation (3.20) implies that the characteristic sequence $u^p[n]$ is readily calculated by the DFT of the sampled B-splines. On the other hand, once the characteristic

Table 3.1 Values of the B-splines B^p at grid points

k	-4	-3	-2	-1	0	1	2	3	4
$B^2(k)$	0	0	0	0	1	0	0	0	0
$B^3(k) \times 8$	0	0	0	1	6	1	0	0	0
$B^4(k) \times 6$	0	0	0	1	4	1	0	0	0
$B^5(k) \times 384$	0	0	1	76	230	76	1	0	0
$B^6(k) \times 120$	0	0	1	26	66	26	1	0	0
$B^7(k) \times 2^6\,6!$	0	1	722	10,543	23,548	10,543	722	1	0
$B^8(k) \times 7!$	0	1	120	1191	2,416	1,191	120	1	0
$B^9(k) \times 2^8\,7!$	1	6,552	331,610	2485,288	4,675,014	2,485,288	331,610	6,552	1
$B^{10}(k) \times 9!$	1	502	14,608	88,234	156,190	88,234	14,608	502	1

function $\zeta^p(v, 0)$ is known, the samples of the B-spline B^p are calculated by the IDFT:

$$B^p(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} e^{2\pi i n k/N}\, u^p[n]. \tag{3.21}$$

Table 3.1 provides samples of B-splines $B^p(t)$ of orders from 2 to 10 at grid points. The samples of higher order splines were computed by the MATLAB function `bspuvP.m`. They are gathered in the file `BSUV.mat`. The sequences $u^p[n]$ are calculated by the MATLAB function `juviP.m`.

Examples of characteristic sequences:

Denote $y = \cos \pi n/N$. Then

$$u_2^p = 1, \quad u^3[n] = \frac{1+y^2}{2}, \quad u^4[n] = \frac{1+2y^2}{3}.$$

$$u^5[n] = \frac{5+18y^2+y^4}{24}, \quad u^6[n] = \frac{2+11y^2+2y^4}{15},$$

$$u^7[n] = \frac{y^6+179y^4+479y^2+61}{720}, \; u^8[n] = \frac{16y^6+4740y^4-4635y^2+1139}{1260}.$$

Figure 3.4 displays the characteristic sequences of several spline spaces (MATLAB code `jusplsP.m`). The characteristic sequences of any order are calculated by MATLAB function `juviP.m`.

In all the above examples the sequences $u^p[n]$ are cosine polynomials $u_n^p = P_r(\cos \pi n/N)$ of degree $r = [(p+1)/2]$ with real coefficients, which have only even degree terms. They are N-periodic, strictly positive and symmetric about $N/2$. This is true in general case.

Proposition 3.4 ([8]) *For any natural p, the sequences $u_n^p = P_r(\cos \pi n/N)$, $r = [(p-1)/2]$, are strictly positive on the real axis. Their maximal values, which are equal to 1, are achieved at points $\{nN\}$, $n \in \mathbb{Z}$, while the minimal values are achieved at points $\{(2n+1)N/2\}$, $n \in \mathbb{Z}$.*

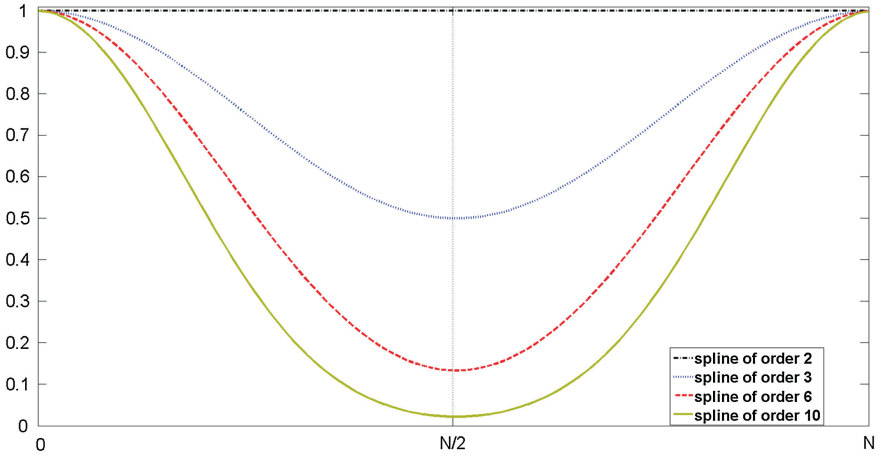


Fig. 3.4 Characteristic sequences of the spaces ${}^{p}\mathscr{S}_T$ of splines of orders 2, 3, 6 and 10

Corollary 3.2 *1. The shifts of the periodic B-spline* $\{B^p(Nt/T-k)\}$, $k = 0, ..., N-1$, $p \in \mathbb{N}$, *form a basis of the space* ${}^{p}\mathscr{S}_T$.

2. Any N-periodic signal $\boldsymbol{x} = \{x[k]\}$, $k \in \mathbb{Z}$ *can be uniquely interpolated by a spline from* ${}^{p}\mathscr{S}_T$: $S(Tk/N) = x[k]$.

3.3 Periodic Discrete–Discrete Convolution and Zak Transform

Assume, $N = 2^j$ and $M = 2^m$, where j and $m < j$ are natural numbers. Denote $N_m \stackrel{\text{def}}{=} N/2^m = 2^{j-m}$. Let $\mathbf{f} = \{f[k]\} \in \Pi[N]$ be an N-periodic signal and $\mathbf{q} = \{q[k]\} \in \Pi[N_m]$ be an N_m-periodic signal. The signal

$$\mathbf{q} \circledast \mathbf{f}_{[m]} \stackrel{\text{def}}{=} \left\{ \sum_{l=0}^{N_m-1} q[l] f[k-Ml] \right\}, \; k \in \mathbb{Z}, \tag{3.22}$$

is referred to as the periodic M-fold mixed discrete-discrete circular convolution. Denote by $\mathscr{M}[\mathbf{f}_{[m]}]$ the subspace of $\Pi[N]$, which consists of M-fold mixed periodic convolutions of different signals $\mathbf{q} \in \Pi[N_m]$ with the signal $\mathbf{f} \in \Pi[N]$:

$$\mathbf{g} \in \mathscr{M}[\mathbf{f}_{[m]}] \Longleftrightarrow g[k] = \sum_{l=0}^{N_m-1} q[l] f[k-Ml], \tag{3.23}$$

where $\mathbf{q} \in \Pi[N_m]$. The signal $\mathbf{f} \in \Pi[N]$ is called the generator of the space $\mathscr{M}[\mathbf{f}_{[m]}]$.

Remark 3.3.1 The mixed circular convolution defined by Eq. (3.22) is the regular N-periodic discrete circular convolution of the signal $\mathbf{f} \in \Pi[N]$ with the signal $\mathbf{q} \in \Pi[N_m]$, which is upsampled by factor of M.

Equation (1.23) implies that

$$f[k+lM] = \frac{1}{N}\sum_{n=0}^{N-1} \hat{f}[n]\, e^{2\pi i n(k+lM)/N},$$

where $\hat{f}[n] = \sum_{k=0}^{N-1} f[k]\, e^{-2\pi i n k/N}$ is the DFT of the N-periodic signal $\mathbf{f}$. Substituting $n = \nu + rN_m,\ \nu = 0, ..., N_m - 1$, we get

$$\begin{aligned} f[k+lM] &= \frac{1}{N_m}\sum_{\nu=0}^{N_m-1} \frac{1}{M}\sum_{r=0}^{M-1} \hat{f}[\nu + rN_m]\, e^{2\pi i(\nu+rN_m)(k+lM)/N} \\ &= \frac{1}{N_m}\sum_{\nu=0}^{N_m-1} e^{2\pi i \nu l/N_m}\, Z[f_{[m]}](\nu)[k], \end{aligned} \tag{3.24}$$

where for $\nu = 0, ..., N_m - 1$

$$Z[f_{[m]}](\nu)[k] \stackrel{\text{def}}{=} \frac{1}{M}\sum_{r=0}^{M-1} \hat{f}[\nu + rN_m]\, e^{2\pi i(\nu+rN_m)k/N},\ k \in \mathbb{Z}. \tag{3.25}$$

On the other hand, Eq. (3.24) implies that , $f[k+Ml]$ is the IDFT of the N_m-periodic with respect to ν signal $Z[f_{[m]}](\nu)[k]$. Thus,

$$Z[f_{[m]}](\nu)[k] = \sum_{l=0}^{N_m-1} e^{-2\pi i \nu l/N_m}\, f[k+Ml] = \sum_{l=0}^{N_m-1} e^{2\pi i \nu l/N_m}\, f[k-Ml]. \tag{3.26}$$

Definition 3.3 The signal $Z[f_{[m]}](\nu)[k]$, which is defined by Eq. (3.25) or by equivalent Eq. (3.26), is referred to as the M-fold periodic discrete Zak transform.

Equations (3.25) and (3.26) imply that

1. For any fixed value k, the sequence $Z[f_{[m]}](\nu)[k]$ is N_m-periodic with respect to the variable ν.
2. For any fixed value of ν, the sequence $Z[f_{[m]}](\nu)[k]$ is N-periodic with respect to the variable k and belongs to $\mathscr{M}[\mathbf{f}_{[m]}]$.
3. Assume that $d \in \mathbb{Z}$. Then,

$$Z[f_{[m]}](\nu)[k+dM] = e^{2\pi i \nu d/N_m}\, Z[f_{[m]}](\nu)[k]. \tag{3.27}$$

4. Equation (3.25) provides the DFT expansion of the function $Z[f_{[m]}](\nu)[k]$ in the space $\Pi[N]$ of N-periodic discrete signals. Thus, the DFT coefficients

$$\widehat{Z[f_{[m]}]}(\nu)[k][n] = N_m \begin{cases} \hat{f}[n], & \text{if } n = \nu + rN_m, \\ 0, & \text{otherwise,} \end{cases} \tag{3.28}$$
$$= N_m \delta[\nu - n] \,(\text{mod}\, N_m)\ \hat{f}[n].$$

Combining different representations (Eqs. (3.25) and (3.26)) of $Z[f_{[m]}](\nu)[k]$, we get the relations for $n = -N/2, ..., N/2 - 1$:

$$Z[f_{[m]}](\nu)[k] = \frac{1}{M} \sum_{r=0}^{M-1} \hat{f}\,[\nu + rN_m]\ e^{2\pi i(\nu + rN_m)k/N} \tag{3.29}$$
$$= \sum_{l=0}^{N_m-1} e^{-2\pi i\nu l/N_m}\ f[k + Ml].$$

Equation (3.29) provides DFT of polyphase components of the signal $\mathbf{f} \in \Pi[N]$. It implies the following special cases:

$$Z[f_{[m]}](0)[k] = \frac{1}{M} \sum_{r=0}^{M-1} \hat{f}\left[\frac{rN}{M}\right] e^{2\pi irk/M} = \sum_{l=0}^{N_m-1} f[k + Ml], \tag{3.30}$$

$$Z[f_{[m]}](\nu)[0] = \frac{1}{M} \sum_{r=0}^{M-1} \hat{f}\left[\nu + \frac{rN}{M}\right] = \sum_{l=0}^{N_m-1} e^{-2M\pi i\nu l/N}\ f[Ml]. \tag{3.31}$$

Note that Eq. (1.27) is a special case of Eq. (3.29) when $M = 2$.

The sequence $Z[\mathbf{f}_{[m]}(v)[0]$ is called the characteristic sequence of the space $\mathscr{M}[\mathbf{f}_{[m]}]$. The set $\left\{Z[f_{[m]}](\nu)[k]\right\}$ provides a proper framework for the interpolation.

Proposition 3.5 *Assume the N_m-periodic sequence $\left\{Z[f_{[m]}](\nu)[0]\right\}$, $\nu \in \mathbb{Z}$, does not vanish. Then, any signal $\boldsymbol{x} \in \Pi[N_m]$ can be uniquely interpolated by an element $\boldsymbol{g} \in \mathscr{M}[\mathbf{f}_{[m]}]$, that is $g[Mk] = x[k]$, $k = 0, ..., N_m - 1$.*

Proof Let $g[k] = \sum_{l=0}^{N_m-1} q[l] f[k - Ml]$. Then, $g[Mk] = \sum_{l=0}^{N_m-1} q[l] f[M(k-l)]$ is a discrete circular convolution of the signal $\mathbf{q} \in \Pi[N_m]$ with the sampled signal $\{f[Mk]\}$, $k = 0, ...N_m - 1$. Applying the DFT to both parts of the relation $g[Mk] = x[k]$, $k = 0, ..., N - 1$, we get, due to Eq. (1.25),

$$\hat{q}[\nu]_m \sum_{l=0}^{N-1} f[Ml]\, e^{-2\pi i \nu l/N_m} = \hat{x}[n]_m \Longrightarrow \hat{q}[\nu]_m = \frac{\hat{x}[\nu]_m}{Z[f_{[m]}](\nu)[0]}, \quad (3.32)$$

$$q[l] = \frac{M}{N} \sum_{\nu=0}^{N_m-1} e^{2\pi i \nu l/N_m}\, \hat{q}[\nu]_m.$$

■

Corollary 3.3 *If the sequence* $\left\{Z[f_{[m]}](\nu)[0]\right\}$, $\nu \in \mathbb{Z}$, *does not vanish then the space* $\mathscr{M}[\mathbf{f}_{[m]}]$ *is* N_m*-dimensional. Consequently, the signals*

$$\mathbf{f}_l = \{f_l[k]\}, \ f_l[k] \stackrel{\text{def}}{=} f[k - lM], \ k \in \mathbb{Z}, \ \ l = 0, ..., N_m - 1,$$

are linearly independent and form a basis of $\mathscr{M}[\mathbf{f}_{[m]}]$. *In addition, the* N_m*-periodic signals*

$$\mathbf{f}_l^M = \{f_l[Mk]\}, \ l = 0, ..., N_m - 1, \ k \in \mathbb{Z},$$

form a basis of the space $\Pi[N_m]$.

Proposition 3.6 *If the sequence* $\left\{Z[f_{[m]}](\nu)[0]\right\}$, $\nu \in \mathbb{Z}$, *does not vanish then signals* $\mathbf{Z}^\nu[f] \stackrel{\text{def}}{=} \left\{Z[f_{[m]}](\nu)[k]\right\}$, $\nu = -N/2M, ..., N/2M - 1$, $k \in \mathbb{Z}$, *form an orthogonal in the space* $\Pi[N]$ *basis of the space* $\mathscr{M}[\mathbf{f}_{[m]}]$.

Proof Equations (1.24) and (3.28) imply that the inner product in $\Pi[N_m]$

$$\left\langle \mathbf{Z}^\nu[f], \mathbf{Z}^\mu \right\rangle = \sum_{k=0}^{N-1} Z[f_{[m]}](\nu)[k]\, Z^*[f_{[m]}](\mu)[k] \quad (3.33)$$

$$= \frac{1}{N} \sum_{m=0}^{N-1} \widehat{Z[f_{[m]}]}(\nu)[m] \left(\widehat{Z[f_{[m]}]}(\mu)[m]\right)^* = \delta[\nu - \mu]\, N_m \sum_{r=0}^{M-1} |\hat{f}[\nu + rM]|^2.$$

Assume $g[k] = \sum_{l=0}^{N_m-1} q[l] f[k - Ml]$. Equation (3.24) implies that

$$g[k] = \frac{1}{N_m} \sum_{l=0}^{N_m-1} q[l] \sum_{\nu=0}^{N_m-1} e^{-2\pi i \nu l/N_m} Z[f_{[m]}](\nu)[k] = \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} \xi_\nu\, Z[f_{[m]}](\nu)[k],$$

$$\xi_\nu = \hat{q}[\nu]_m = \sum_{l=0}^{N_m-1} q[l]\, e^{-2\pi i \nu l/N_m} \Longrightarrow q[l] = \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} \xi_\nu\, e^{2\pi i \nu l/N_m}.$$

Thus, coordinates of different representations of a signal belonging to the space $\mathscr{M}[\mathbf{f}_{[m]}]$ are linked to each other via the direct and inverse DFT. ■

3.4 A Leading Example: Periodic Discrete Splines

As before, $N = 2^j$, $M = 2^m$, $m < j$, and $N_m = 2^{j-m}$.

3.4.1 Non-periodic Discrete B-Splines

The non-periodic signal

$$\check{\mathbf{b}}^1_{[m]} = \left\{\check{b}^1_{[m]}[k]\right\}, \quad \check{b}^1_{[m]}[k] \stackrel{\text{def}}{=} \begin{cases} 1/M, & \text{if } k = 0, ...M-1; \\ 0, & \text{otherwise.} \end{cases}$$

is called the discrete B-spline of first order. The integer $M = 2^m$ is called its span. The discrete B-splines of higher orders are defined via iterated discrete convolution: $\check{\mathbf{b}}^p_{[m]} \stackrel{\text{def}}{=} \check{\mathbf{b}}^1_{[m]} \star \check{\mathbf{b}}^{p-1}_{[m]}$. The B-spline $\check{\mathbf{b}}^p_{[m]}$ is a finite length signal, which is supported on the set 0, ..., $p(2^m - 1)$.

Remark 3.4.1 The term discrete B-spline is used in literature for designation of different objects. For example in [1, 11, 13] the discrete B-spline is understood as a sampled polynomial B-spline. Our definition of the discrete B-spline is that used in [4, 9, 14].

Denote by

$$k_+^{[r]} \stackrel{\text{def}}{=} \begin{cases} k(k+1)\cdots k+r-1), & \text{if } k = 0, 1, ...; \\ 0, & \text{otherwise.} \end{cases} \tag{3.34}$$

the truncated factorial polynomial.

Proposition 3.7 ([3, 5]) *The discrete B-spline $\check{\mathbf{b}}^p_{[m]}$ of order p is the piecewise sampled polynomial of degree $p-1$*

$$\check{b}^p_{[m]}[k] = \frac{1}{M^p(p-1)!}\sum_{l=0}^{p}(-1)^p\binom{p}{l}(k+1-lM)_+^{[p-1]} \tag{3.35}$$

whose breakpoints (nodes) are $\{lM\}$, $l \in \mathbb{Z}$.

It is positive within its support and symmetric about $p(M-1)/2$. The B-spline $\check{b}^1_{[m]}[k]$ increases monotonically as $k < p(M-1)/2$ and decays as $k > p(M-1)/2$.

3.4.2 Periodic Discrete B-Splines

Periodic discrete B-splines $\tilde{\mathbf{b}}^1_{[m]}$ of first order are derived by N-periodization of the B-splines $\check{\mathbf{b}}^1_{[m]}$: $\tilde{b}^1_{[m]}[k] \stackrel{\text{def}}{=} \sum_{l\in\mathbb{Z}} f\check{b}^1_{[m]}[k+lN]$. The DFT of the B-splines $\tilde{\mathbf{b}}^1_{[m]}$

$$\hat{\tilde{b}}^{1}_{[m]}[n] = \frac{1}{M}\sum_{k=0}^{M-1}\omega^{-kn} = \frac{1-\omega^{-Mn}}{M(1-\omega^{-n})}, \quad \omega = e^{2\pi i/N}.$$

The discrete B-splines of higher orders are defined via iterated discrete circular convolution: $\tilde{\mathbf{b}}^{p}_{[m]} \stackrel{\text{def}}{=} \tilde{\mathbf{b}}^{1}_{[m]} \circledast \tilde{\mathbf{b}}^{p-1}_{[m]}$. Consequently, the DFT

$$\hat{\tilde{b}}^{p}_{[m]}[n] = \left(\frac{1-\omega^{-Mn}}{M(1-\omega^{-n})}\right)^{p}.$$

Certainly, as $k = 0, ..., N-1$, the periodic B-spline $\tilde{b}^{p}_{[m]}[k]$ coincides with the non-periodic one $\check{b}^{p}_{[m]}[k]$.

In the sequel, we deal with the discrete splines of even orders $p = 2r$. The B-splines of even order can be centered:

$$\mathbf{b}^{2r}_{[m]} = \left\{b^{2r}_{[m]}[k]\right\}, \quad b^{2r}_{[m]}[k] = \tilde{b}^{2r}_{[m]}[k + r(2^m - 1)].$$

The DFT of the centered periodic discrete B-splines

$$\hat{b}^{2r}_{[m]}[n] = \sum_{k=0}^{N-1}\omega^{-kn}\, b^{2r}_{[m]}[k] = \left(\frac{\sin(2^m\pi\, n/N)}{2^m\, \sin(\pi\, n/N)}\right)^{2r}, \quad \hat{b}^{2r}_{[m]}[0] = 1. \tag{3.36}$$

Thus,

$$b^{2r}_{[m]}[k] = \frac{1}{N}\sum_{n=0}^{N-1}\omega^{kn}\,\hat{b}^{2r}_{[m]}[n] = \frac{1}{N}\sum_{n=0}^{N-1}\omega^{kn}\,\left(\frac{\sin(2^m\pi\, n/N)}{2^m\, \sin(\pi\, n/N)}\right)^{2r}. \tag{3.37}$$

Examples

$$\hat{\mathbf{b}}^{2r}_{[2]}[n] = \cos^{2r}\frac{\pi\, n}{N}, \quad \hat{\mathbf{b}}^{2r}_{[4]}(v) = \left(\cos\frac{\pi\, n}{N}\,\cos\frac{2\pi\, n}{N}\right)^{2r}. \tag{3.38}$$

Remark 3.4.2 We emphasize that the values $b^{2r}_{[m]}[k]$ are non-negative integers. It is worth noting that the discrete B-spline $\mathbf{b}^{2r}_{[m]}$ is not a sampled polynomial B-spline.

3.4.3 Spaces of Periodic Discrete Splines

A signal $\mathbf{s}^{2r}_{[m]} = \left\{s^{2r}_{[m]}[k],\ k \in \mathbb{Z}\right\}$, which can be represented as the M-fold mixed periodic discrete-discrete convolution of a signal $\mathbf{q} \in \Pi[N_m]$ with a periodic discrete B-spline

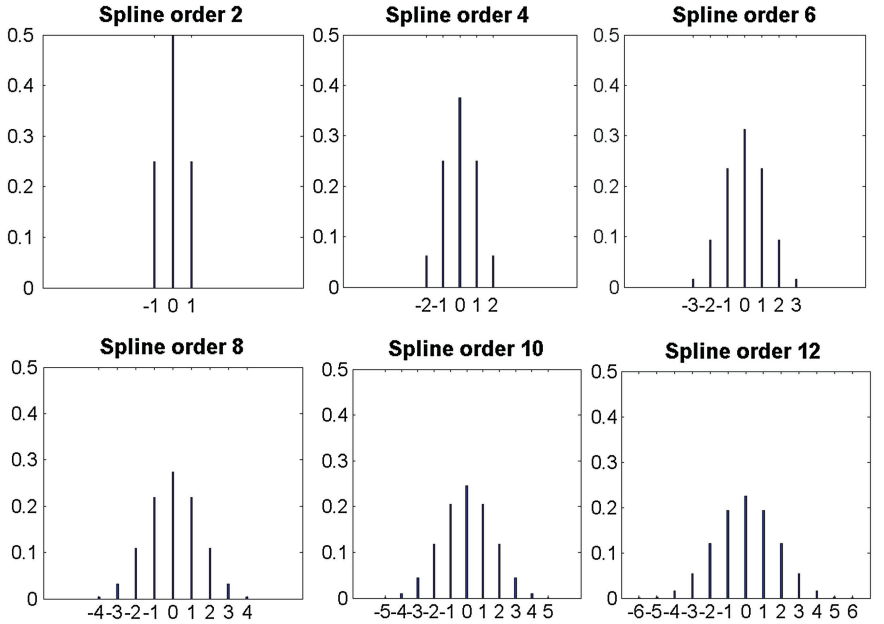


Fig. 3.5 Centered B-splines $\mathbf{b}_{[2]}^{2r}$ of orders 2, 4, 6, 8, 10 and 12

$$\mathbf{s}_{[m]}^{2r} = \mathbf{q} \circledast (\mathbf{b}_{[m]}^{2r})_M \Longleftrightarrow s_{[m]}^{2r}[k] = \sum_{l=0}^{N_m-1} q[l]\, b_{[m]}^{2r}[k - Ml], \quad M = 2^m, \quad (3.39)$$

is called a periodic discrete spline. The space of the periodic discrete splines defined by Eq. (3.39) is denoted ${}^{2r}\mathscr{S}_{[m]}$.

The Zak transform of the periodic discrete B-spline is

$$\zeta_{[m]}^{2r}(\nu)[k] = Z[\mathbf{b}_{[m]}^{2r}](\nu)[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N_m-1} e^{2\pi i \nu l/N_m}\, b_{[m]}^{2r}[k - Ml] \quad (3.40)$$

$$= \frac{1}{M} \sum_{\mu=0}^{M-1} \hat{b}_{[m]}^{2r}[\nu + \mu N_m]\, e^{2\pi i(\nu+\mu N_m)k/N_m} \quad (3.41)$$

$$= \frac{1}{M} \sum_{\mu=0}^{M-1} \omega^{k(\nu+N_m\mu)} \left(\frac{\sin(2^m\, \pi\, (\nu + N_m\mu))/N)}{2^m\, \sin(\pi(\nu + N_m\mu)/N)} \right)^{2r}.$$

Thus, the DFT of $\zeta_{[m]}^{2r}(\nu)[k]$ in the space $\Pi[N]$ comprises only 2^m terms

$$\hat{\zeta}^{2r}_{[m]}(\nu)[n] = N_m\, \hat{b}^{2r}_{[m]}[n]\, \delta[\nu - n](\mathrm{mod}\, N_m) \tag{3.42}$$
$$= N_m \left(\frac{\sin 2^m \pi\, n/N)}{2^m\, \sin \pi n/N)} \right)^{2r} \delta[\nu - n](\mathrm{mod}\, N_m)$$

The set of N-periodic with respect to k discrete splines $\zeta^{2r}_{[m]}(\nu) = \left\{ \zeta^{2r}_{[m]}(\nu) \right\} \in {}^{2r}\mathscr{S}_{[m]}$ is N_m-periodic with respect to ν. We call them the discrete exponential splines. Equation (3.40) implies that

$$b^{2r}_{[m]}[k - Ml] = \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} e^{-2\pi i \nu l/N_m}\, \zeta^{2r}_{[m]}(\nu)[k],$$
$$s^{2r}_{[m]}[k] = \sum_{l=0}^{N_m-1} q[l]\, b^{2r}_{[m]}[k - Ml] = \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} \hat{q}[\nu]_m\, \zeta^{2r}_{[m]}(\nu)[k].$$

The characteristic sequence of the space ${}^{2r}\mathscr{S}_{[m]}$

$$U^{2r}_{[m]}[\nu] \stackrel{\mathrm{def}}{=} \zeta^{2r}_{[m]}(\nu)[0] = \sum_{l=0}^{N_m-1} \omega^{-2^m \nu l}\, b^{2r}_{[m]}[2^m l] \tag{3.43}$$
$$= \frac{1}{M} \sum_{n=0}^{2^m-1} \hat{b}^{2r}_{[m]}[\nu + N_m n]$$
$$= \frac{1}{M} \sum_{n=0}^{M-1} \left(\frac{\sin(2^m \pi\, (\nu + N_m n)/N)}{2^m\, \sin(\pi(\nu + N_m n)/N)} \right)^{2r}$$
$$= \frac{\sin^{2r}(\pi\, \nu/N_m)}{2^m} \sum_{l=0}^{M-1} \frac{1}{(2^m\, \sin(\pi(\nu + N_m l)/N))^{2r}}.$$

It is readily seen that $U^{2r}_{[m]}[\nu]$ is strictly positive.

Proposition 3.8 *1. The discrete exponential splines* $\left\{ \zeta^{2r}_{[m]}(\nu)[k] \right\}$, $\nu = -N_m/2, ..., N_m/2 - 1$, *form an orthogonal basis of the space* ${}^{2r}\mathscr{S}_{[m]}$.

2. Their squared norms in the space $\Pi[N]$ *are*

$$\left\| \zeta^{2r}_{[m]}(\nu)[k] \right\|^2 = N_m\, U^{4r}_{[m]}[\nu]. \tag{3.44}$$

3. The circular convolution of two discrete exponential splines is a discrete exponential spline:

$$\zeta^{2r}_{[m]}(\nu) \circledast \zeta^{2q}_{[m]}(\mu)[k] = N_m\, \delta[\nu - \mu]\, \zeta^{2(r+q)}_{[m]}(\nu)[k]. \tag{3.45}$$

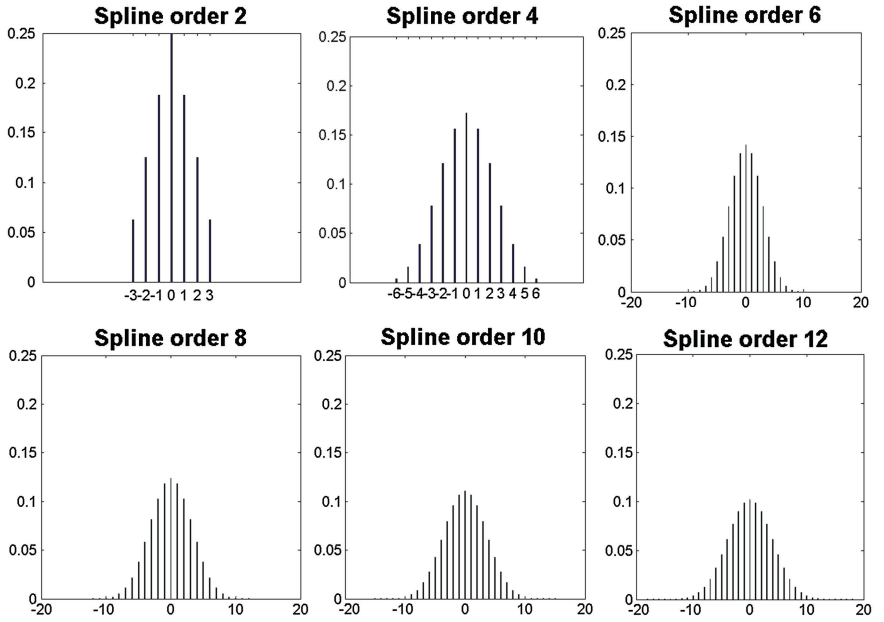


Fig. 3.6 Centered B-splines $\mathbf{b}_{[4]}^{2r}$ of orders 2, 4, 6, 8, 10 and 12

4. *The discrete exponential splines are the eigenvectors of the shift operator:*

$$\zeta_{[m]}^{2r}(\nu)[k + dM] = e^{2\pi i \nu d/N_m} \zeta_{[m]}^{2r}(\nu)[k] \Longrightarrow \tag{3.46}$$

$$\zeta_{[m]}^{2r}(\nu)[lM] = e^{2\pi i \nu l/N_m}\, U_{[m]}^{2r}[\nu], \quad \frac{\zeta_{[m]}^{2r}(\nu)[lM]}{U_{[m]}^{2r}[\nu]} = e^{2\pi i \nu l/N_m}. \tag{3.47}$$

Equation (3.47) means that the discrete exponential splines interpolate the Fourier exponential functions.

5. *Any N_m-periodic signal $\boldsymbol{x} \in \Pi[N_m]$ can be interpolated by a spline $\boldsymbol{s}_{[m]}^{2r} \in {}^{2r}\mathscr{S}_{[m]}$: $s_{[m]}^{2r}[Ml] = x[l],\ k = 0, ..., N_m - 1$, where*

$$s_{[m]}^{2r}[k] = \frac{1}{N_m} \sum_{\nu=0}^{N-1} \frac{\hat{\mathbf{x}}[n]_m}{U_{[m]}^{2r}[\nu]}\, \zeta_{[m]}^{2r}(\nu)[k]. \tag{3.48}$$

Proof

1. It is a direct consequence of Proposition 3.6.
2. Equations (3.33) and (3.43) imply that

$$\left\| \zeta^{2r}_{[m]}(\nu)[k] \right\|^2 = N_m \sum_{r=0}^{M-1} \left| \hat{b}^{2r}_{[m]}[\nu + rN] \right|^2$$
$$= \frac{N_m}{M} \sum_{n=0}^{M-1} \left(\frac{\sin(2^m \pi\,(\nu + N_m n)/N)}{2^m\ \sin(\pi(\nu + N_m n)/N)} \right)^{4r} = N_m U^{2p}_{[m]}[\nu].$$

3. Let $g[k] = \zeta^{2r}_{[m]}(\nu) \circledast \zeta^{2q}_{[m]}(\mu)[k]$ be the circular convolution of two discrete exponential splines. Then the DFT coefficients of $\mathbf{g}$ are the products

$$\hat{g}[n] = \hat{\zeta}^{2r}_{[m]}(\nu)[n]\ \hat{\zeta}^{2q}_{[m]}(\mu)[n]$$
$$= N_m^2 \delta[\nu - n](\mathrm{mod} N_m)\ \delta[\mu - n](\mathrm{mod} N_m) \left(\frac{\sin 2^m \pi\, n/N}{2^m\ \sin \pi n/N} \right)^{2(r+q)}$$
$$= \delta[\mu - n]\, N_m \hat{\zeta}^{2(r+q)}_{[m]}(\nu)[n].$$

4. This is a special case of Eq. (3.27).
5. It is a direct consequence of Proposition 3.5.

■

References

1. A. Aldroubi, M. Eden, M. Unser, Discrete spline filters for multiresolutions and wavelets of l_2. SIAM J. Math. Anal. **25**(5), 1412–1432 (1994)
2. M.J. Bastiaans, in *Gabor's Expansion and the Zak Transform for Continuous-Time and Discrete-Time Signals*, ed. by Y.Y. Zeevi, R. Coifman. Signal and Image Representation in Combined Spaces, vol 7 (Academic Press, San Diego, 1998), pp. 23–69 (Wavelet Anal. Appl.)
3. K. Ichige, M. Kamada, An approximation for discrete B-splines in time domain. IEEE Signal Process. Lett. **4**(3), 82–84 (1997)
4. V.N. Malozemov, A.B. Pevnyi, Discrete periodic splines and their numerical applications. Comput. Math. Math. Phys. **38**(8), 1181–1192 (1998)
5. A.B. Pevnyi, V. Zheludev, On the interpolation by discrete splines with equidistant nodes. J. Approx. Theor. **102**(2), 286–301 (2000)
6. M. Pinsky, in *Introduction to Fourier Analysis and Wavelets* (Brooks Cole, Salt Lake City, 2002)
7. A. Ron, in *Introduction to Shift-Invariant Spaces: Linear Independence*, ed. by A. Pinkus, D. Leviatan, N. Din, D. Levin. Multivariate Approximation and Applications (Cambridge University Press, Cambridge, 2001), pp. 112–151
8. I.J. Schoenberg, Cardinal interpolation and spline functions. J. Approximation Theor. **2**, 167–206 (1969)
9. L.L. Schumaker, *Spline Functions: Basic Theory* (Wiley, New York, 1981)
10. E. Stein, G. Weiss, in *Introduction to Fourier Analysis on Euclidean Spaces* (Princeton University Press, Princeton, 1971)
11. M. Unser, A. Aldroubi, M. Eden, Fast B-spline transforms for continuous image representation and interpolation. IEEE Trans. Pattern Anal. Mach. Intell. **13**(3), 277–285 (1991)
12. M. Unser, A. Aldroubi, M. Eden, On the asymptotic convergence of B-spline wavelets to Gabor functions. IEEE Trans. Inform. Theor **38**(2), 864–872 (1992)

13. M. Unser, A. Aldroubi, M. Eden, B-Spline Signal processing. I. Theory. IEEE Trans. Signal Process., **41**(2), 821–833 (1993) (IEEE Signal Processing Society's 1995 best paper award)
14. K.F. Üstüner, L.A. Ferrari, Discrete splines and spline filters. IEEE Trans. Circ. Syst. II: Analog Digit. Sig. Process. **39**(7), 417–422 (1992)
15. J. Zak, Finite translations in solid-state physics. Phys. Rev. Lett. **19**(24), 1385–1387 (1967)

Chapter 4
Periodic Polynomial Splines

Abstract In this chapter the spaces of periodic polynomial splines, which are introduced in Sect. 3.2.2, are discussed in more details. It is shown that the periodic exponential splines generate a specific form of harmonic analysis in these spaces. A family of generators of the spaces of periodic splines is presented.

4.1 Spline Harmonic Analysis (SHA) in the Space of Periodic Polynomial Splines

The exponential splines defined in Sect. 3.2.2 provide a flexible computational scheme in the space of periodic polynomial splines.

4.1.1 Orthogonal Periodic Exponential Splines

We recall that $N = 2^j,\ j \in \mathbb{N}$ and $\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$. The centered N-periodic B-spline $B^p(t)$ of order p on the grid $\{k\},\ k \in \mathbb{Z}$, is defined in Sect. 3.2.2. The space ${}^p\mathcal{S}_T$ consists of T-periodic splines

$$S^p\left(\frac{N}{T}\right) = \sum_{k=0}^{N-1} q[k] B^p\left(\frac{Nt}{T} - k\right) \tag{4.1}$$

on the grid $\{Tk/N\},\ k \in \mathbb{Z}$. For simplicity, we assume that T=N, thus the splines are defined on the grid $\{k\},\ k \in \mathbb{Z}$, and ${}^p\mathcal{S} \stackrel{\text{def}}{=} {}^p\mathcal{S}_N$. Rescaling to the space ${}^p\mathcal{S}_T$ is straightforward.

The Zak transform of the periodic B-spline, which we call the periodic exponential spline, was defined in Eq. (3.17):

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_4,

$$\zeta^p[n](t) \stackrel{\text{def}}{=} Z[B^p][n](t) = \sum_{k=0}^{N-1} \omega^{nk} B^p(t-k) \tag{4.2}$$
$$= \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+l)t} \left(\frac{\sin\pi(n/N+l)}{\pi(n/N+l)}\right)^p.$$

Equation (4.2) provides expansion of the N-periodic function $\zeta^p[n](t)$ into the Fourier series. Thus, the Fourier coefficients

$$c_{n+Nl}(\zeta^p[n]) = N\left(\frac{\sin\pi(n/N+l)}{\pi(n/N+l)}\right)^p, \quad l\in\mathbb{Z}. \tag{4.3}$$

The rest of the Fourier coefficients are zero. This can be formalized as follows

$$c_k(\zeta^p[n]) = N\delta[k-n](\text{mod}N)\left(\frac{\sin\pi k/N}{\pi k/N}\right)^p, \quad k\in\mathbb{Z}. \tag{4.4}$$

where $\delta[k]$ is the Kronecker delta.

Denote the characteristic sequence of the space ${}^p\mathscr{S}$

$$u^p[n] \stackrel{\text{def}}{=} \zeta^p[n](0) = \sum_{k=0}^{N-1} \omega^{-nk} B^p(k) \tag{4.5}$$
$$= \sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi(n/N+l)}{\pi(n/N+l)}\right)^p$$
$$= \sin^p\frac{\pi n}{N} \sum_{l\in\mathbb{Z}} \left(\frac{(-1)^l}{\pi(n/N+l)}\right)^p.$$

The sequence $u^p[n]$ is the DFT of the sampled periodic B-spline $B^p(t)$. It is N-periodic and symmetric about zero, where it attains its maximal value $u^p[0]=1$. The minimal value is attained at $n = N/2$ and it is positive. The characteristic sequences of several spline spaces are displayed in Fig. 3.4.

We summarize the properties of the periodic exponential splines.

Proposition 4.1. *1. The exponential spline $\zeta^p[n](t)$, is orthogonal to all the exponential splines $\zeta^q[m](t)$ where q is an arbitrary natural number and $m\neq n$.*

2. The exponential splines $\{\zeta^p[n](t)\}$, $n=-N/2, ..., N/2-1$, form an orthogonal basis of the space ${}^p\mathscr{S}$. If a spline $S^p(t)\in{}^p\mathscr{S}$ is defined by Eq. (4.1), then it can be represented as

$$S^p(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n]\,\zeta^p[n](t), \quad \xi[n] \stackrel{\text{def}}{=} \hat{q}[n]. \tag{4.6}$$

3. *Squared norms of the exponential splines are*

$$\|\zeta^p[n]\|^2 = N \sum_{l\in\mathbb{Z}} \left(\frac{\sin \pi\,(n/N + l)}{\pi\,(n/N + l)} \right)^{2p} = N u^{2p}[n]. \tag{4.7}$$

4. *Derivatives of order s of the exponential splines of order p are exponential splines of order $p - s$:*

$$\zeta^p[n](t)^{(s)} = (1 - \omega^{-n})^s \zeta^{p-s}[n] \left(t + \frac{s}{2} \right). \tag{4.8}$$

5. *Derivatives of the exponential spline $\zeta^p[n](t)^{(s)}$, are orthogonal to all the derivatives $\zeta^q[m](t)^{(r)}$ of the exponential splines where $q, s < p$, $r < q$ and $m \neq n$.*
6. *The squared norms of the derivatives are*

$$\|(\zeta^p[n])^{(s)}\|^2 = N w[n]^{2s}\, u^{2(p-s)}[n], \quad w[n] \stackrel{\text{def}}{=} 2 \sin \frac{\pi n}{N}. \tag{4.9}$$

7. *The circular convolution of two exponential splines is*

$$\zeta^p[n] \circledast \zeta^q[m](t) = N \delta[m - n] \zeta^{p+q}[n](t). \tag{4.10}$$

8. *The exponential splines are eigenfunctions of the integer shift operator:*

$$\zeta^p[n]\,(t + d) = \omega^{nd} \zeta^p[n](t) \Longrightarrow \zeta^p[n]\,(d) = \omega^{nd}\, u^p[n],\ d \in \mathbb{N}. \tag{4.11}$$

9. *The exponential splines are eigenfunctions of the finite differences operators:*

$$\begin{aligned} \Delta_1^s[\zeta^p[n]\,(t)] &= (\omega^n - 1)^s \zeta^p[n](t), \\ \delta_1^{2s}[\zeta^p[n]\,(t)] &= \left(-4 \sin^2 \frac{\pi n}{N} \right)^s \zeta^p[n](t). \end{aligned} \tag{4.12}$$

10. *The exponential splines interpolate the Fourier exponentials:*

$$\frac{\zeta^p[n]\,(d)}{u^p[n]} = e^{2\pi i n d/N},\ d \in \mathbb{N}. \tag{4.13}$$

Proof

1. The claim follows from Eq. (4.4) and the Parseval identity (1.8).
2. This is a direct consequence of Proposition 3.2.
3. Applying the Parseval identity (1.8), we derive Eq. (4.7).
4. Equation (4.2) implies

$$\zeta^p[n] \left(t - \frac{p}{2} \right) = \sum_{l\in\mathbb{Z}} e^{2\pi i (n/N + l)t}\, e^{-\pi i p (n/N + l)} \left(\frac{\sin \pi\,(n/N + l)}{\pi\,(n/N + l)} \right)^p$$

$$= \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+l)t} \left(\frac{1-e^{-2\pi i(n/N+l)}}{2\pi\, i(n/N+l)}\right)^p$$
$$= \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+l)t} \left(\frac{1-e^{-2\pi i n/N}}{2\pi\, i(n/N+l)}\right)^p$$
$$= (1-\omega^{-n})^p \sum_{l\in\mathbb{Z}} \frac{e^{2\pi i(n/N+l)t}}{(2\pi\, i(n/N+l))^p}. \tag{4.14}$$

Thus, the derivative is

$$\zeta^p[n]\left(t-\frac{p}{2}\right)^{(s)} = (1-\omega^{-n})^s\,(1-\omega^{-n})^{p-s} \sum_{l\in\mathbb{Z}} \frac{e^{2\pi i(n/N+l)t}}{(2\pi\, i(n/N+l))^{p-s}} \tag{4.15}$$
$$= (1-\omega^{-n})^s \zeta^{p-s}[n]\left(t-\frac{p-s}{2}\right).$$

Hence, Eq. (4.8) follows.

5. This is a direct consequence of item 4.1 and item 4.1
6. Equation (4.15) provides the Fourier expansion of the derivative. Consequently, using the Parseval identity (1.8) and the fact that for N-periodic function with any$a$, $\int_0^N f(t)\,dt = \int_0^N f(t+a)\,dt$, we get

$$\int_0^N \zeta^p[n](t)^{(s)} \left(\zeta^p[n](t)^{(s)}\right)^* dt = \int_0^N \zeta^p[n]\left(t-\frac{p}{2}\right)^{(s)} \left(\zeta^p[n]\left(t-\frac{p}{2}\right)^{(s)}\right)^* dt$$
$$= \left(2-e^{2\pi i n/N}-e^{-2\pi i n/N}\right)^s \int_0^N \zeta^{p-s}[n]\left(t-\frac{p-s}{2}\right)\left(\zeta^{p-s}[n]\left(t-\frac{p-s}{2}\right)\right)^* dt$$
$$= \left(2\sin\frac{\pi n}{N}\right)^{2s} \int_0^N \zeta^{p-s}[n](t)\,\left(\zeta^{p-s}[n](t)\right)^* dt = N\left(2\sin\frac{\pi n}{N}\right)^{2s} u^{2(p-s)}[n].$$

7. If $g(t) = \zeta^p[n] \circledast \zeta^q[m](t)$ is the circular convolution of two exponential splines then, due to Eqs. (1.9) and (4.3), the Fourier coefficients

$$c_l(g) = \frac{1}{N} c_l(\zeta^p[n])\, c_l(\zeta^p[m]) = \delta[m-n] N \left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^{p+q}.$$

Hence, Eq. (4.10) follows.

8. This is a direct consequence of Eq. (3.4).
9. Equation (4.11) implies that the first order difference

$$\Delta_1^1[\zeta^p[n]\,(t)] \stackrel{\text{def}}{=} \zeta^p[n]\,(t+1) - \zeta^p[n]\,(t) = (\omega^n-1)\zeta^p[n](t)$$
$$\Longrightarrow \Delta_1^s[\zeta^p[n]\,(t)] = (\omega^n-1)^s \zeta^p[n](t).$$

The second order central difference

$$\delta_1^2[\zeta^p[n](t)] \stackrel{\text{def}}{=} \zeta^p[n](t+1) - 2\zeta^p[n](t) + \zeta^p[n](t-1)$$
$$= \left(e^{2\pi in/N} + e^{-2\pi in/N} - 2\right)\zeta^p[n](t) = -4\sin^2\frac{\pi n}{N}\,\zeta^p[n](t).$$

Hence, Eq. (4.12) follows.

10. Equation (4.13) is a special case of Eq. (4.11) when $t = 0$.

■

Remark 4.1.1. We observe from Proposition 4.1 that the properties of the periodic exponential splines $\zeta^p[n](t)$ are similar to the properties of the Fourier exponential functions $e^{2\pi int/T}$, which were listed in Sect. 1.1. The functions $\left\{e^{2\pi int/T}\right\}$, $n \in \mathbb{Z}$, generate a harmonic analysis in the space $\mathscr{P}_T$ of T-periodic signals. Similarly, the splines $\{\zeta^p[n](t)\}$, $n = -N/2, ..., N/2-1$, generate a kind of harmonic analysis in the space ${}^p\mathscr{S}$ of N-periodic splines. This analysis, which is called the Spline Harmonic Analysis (SHA), is presented in the following sections.

4.1.2 Normalized Periodic Exponential Splines

Using Eq. (4.7), we normalize the exponential splines

$$\gamma^p[n](t) \stackrel{\text{def}}{=} \frac{\zeta^p[n](t)}{\|\zeta^p[n]\|} = \frac{\zeta^p[n](t)}{\sqrt{Nu^{2p}[n]}}, \quad \zeta^p[n](t) = \sqrt{Nu^{2p}[n]}\,\gamma^p[n](t). \quad (4.16)$$

The exponential splines $\{\gamma^p[n](t)\}$, $n = -N/2, ..., N/2-1$, form an orthonormal basis of the space ${}^p\mathscr{S}$. If a spline $S^p(t) \in {}^p\mathscr{S}$ is $S^p(t) = \sum_{k=0}^{N-1} q[k]B^p(t-k)$, then it can be represented as

$$S^p(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\xi[n]\,\zeta^p[n](t) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\sigma[n]\,\gamma^p[n](t), \quad (4.17)$$
$$\xi[n] \stackrel{\text{def}}{=} \hat{q}[n], \quad \sigma[n] = \sqrt{N}\left\langle S^p(t), \gamma^p[n](t)\right\rangle = \sqrt{u^{2p}[n]}\,\hat{q}[n].$$

Properties of the normalized periodic exponential splines $\gamma^p[n](t)$ are directly derived from the properties of the exponential splines $\zeta^p[n](t)$, which are listed in Proposition 4.1.

Derivatives:

$$\gamma^p[n](t)^{(s)} = \sqrt{\frac{u^{2(p-s)}[n]}{u^{2p}[n]}}(1-\omega^{-n})^s\gamma^{p-s}[n]\left(t+\frac{s}{2}\right). \quad (4.18)$$

Norms of derivatives:

$$\|(\gamma^p[n])^{(s)}\|^2 = W^{p,s}[n], \quad W^{p,s}[n] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi n}{N}\right)^{2s} \frac{u^{2(p-s)}[n]}{u^{2p}[n]}. \tag{4.19}$$

Integer shift:

$$\begin{aligned} &\gamma^p[n]\,(t+d) = \omega^{nd}\gamma^p[n](t) \Longrightarrow \\ &\gamma^p[n]\,(d) = \frac{\omega^{nd}}{\sqrt{N}}\,V^p[n], \quad V^p[n] \stackrel{\text{def}}{=} \frac{u^p[n]}{\sqrt{u^{2p}[n]}}, \quad d \in \mathbb{N}. \end{aligned} \tag{4.20}$$

Circular convolution:

$$\begin{aligned} &\gamma^p[n] \circledast \gamma^q[m](t) = \delta[m-n]\,\sqrt{N}\,R^{p,q}[n]\,\gamma^{p+q}[n](t), \\ &R^{p,q}[n] \stackrel{\text{def}}{=} \sqrt{\frac{u^{2(p+q)}[n]}{u^{2p}[n]\,u^{2q}[n]}}. \end{aligned} \tag{4.21}$$

4.1.3 Representation of Periodic Splines by Exponential Splines Bases

Expansion of periodic splines over the orthogonal and orthonormal exponential splines bases, as presented in Eqs. (4.6) and (4.17), imposes a specific form of harmonic analysis methodology onto the spline space, which we call the Spline Harmonic Analysis (SHA). The exponential splines act as harmonics. The coordinates $\{\sigma[n]\}$, $n = -N/2, ..., N/2-1$ (Eq. (4.17)), which act as the Fourier coefficients, are referred to as the *SHA-spectrum* of the spline $S^p(t)$. Originally, this construction of the SHA was presented in its final form in [5, 9] but some of its components were exposed in [4, 7, 8]. The usage of SHA significantly simplifies many operations on splines. We now describe basic relations of the SHA.

Differentiation: Let $S^p \in {}^p\mathscr{S}$ be represented as

$$S^p(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n]\zeta^p[n](t) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1} \sigma[n]\gamma^p[n](t). \tag{4.22}$$

Then, Eqs. (4.15) and (4.18) imply

$$S^p(t)^{(s)} = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} (1-\omega^{-n})^s\,\xi[n]\,\zeta^{p-s}[n]\left(t+\frac{s}{2}\right) \tag{4.23}$$

$$= \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sqrt{\frac{u^{2(p-s)}[n]}{u^{2p}[n]}} (1-\omega^{-n})^s \, \sigma[n] \, \gamma^{p-s}[n].$$

Parseval identities: Let $S^p \in {}^p\mathscr{S}$ be represented as in Eq. (4.22) and

$$T^p(t) = \frac{1}{N} \sum_{l=0}^{N-1} \tau[l] \, \zeta^p[l](t) = \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} \theta[l] \, \gamma^p[l](t) \in {}^p\mathscr{S}. \tag{4.24}$$

From Eqs. (4.22), (4.24), (4.9) and (4.19), we get:

$$\begin{aligned} \left\langle S^p, T^p \right\rangle &= \int_0^N S^p(t) \, T^p(t)^* \, dt = \int_0^N \frac{1}{N} \sum_{n,l=0}^{N-1} \sigma[n] \, \theta[l]^* \, \gamma^p[n](t) \;\; \gamma^p[l](t)^* \, dt \\ &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sigma[n] \, \theta[n]^* \int_0^N |\gamma^p[n](\theta)|^2 d\theta = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sigma[n] \, \theta[n]^*, \end{aligned} \tag{4.25}$$

$$\|S^p\|^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |\sigma[n]|^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} u^{2p}[n] \, |\xi[n]|^2, \tag{4.26}$$

$$\|(S^p)^{(s)}\|^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |\xi[n]|^2 \, \tilde{W}^{p,s}[n] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |\sigma[n]|^2 \, W^{p,s}[n] \tag{4.27}$$

$$\tilde{W}^{p,s}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^{2s} u^{2(p-s)}[n], \quad W^{p,s}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^{2s} \frac{u^{2(p-s)}[n]}{u^{2p}[n]}.$$

Circular convolution: Let $S^p \in {}^p\mathscr{S}$ be represented as in Eq. (4.17) and

$$S^q(t) = \frac{1}{N} \sum_{l=-N/2}^{N/2-1} \lambda[l] \, \zeta^q[l](t) = \frac{1}{\sqrt{N}} \sum_{l=-N/2}^{N/2-1} \eta[l] \, \gamma^q[l](t) \in {}^q\mathscr{S}. \tag{4.28}$$

From Eq. (4.10), we have

$$\begin{aligned} S^p \circledast S^q(t) &= \int_0^N S^p(t-\tau) \, S^q(\tau) \, d\tau = \frac{1}{N^2} \sum_{n,l=-N/2}^{N/2-1} \xi[n] \, \lambda[l] \, \zeta^p[n] \circledast \zeta^q[l](t) \\ &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n] \, \lambda[n] \, \zeta^{p+q}[n](t) \\ &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} R^{p,q}[n] \, \sigma[n] \, \eta[n] \, \gamma^{p+q}[n](t) \in {}^{p+q}\mathscr{S}, \end{aligned} \tag{4.29}$$

where $R^{p,q}[n]$ is the sequence defined in Eq. (4.21).
Sampling and interpolation: The samples of the spline $S^p(t)$ are obtained from Eq. (4.11):

$$S^p(k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n]\,\zeta^p[n](k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n]\,u^p[n]\,\omega^{kn} \tag{4.30}$$
$$= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \sigma[n]\,\gamma^p[n](k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{\sigma[n]}{\sqrt{u^{2p}[n]}}\,u^p[n]\,\omega^{kn}.$$

Equation (4.30) means that the samples $\{S^p(k)\},\ k = 0, \ldots N-1$ are the inverse DFT of the sequence $\{\xi[n]\,u^p[n]\}$.
Assume the spline $S^p(t)$ interpolates N-periodic data $\{z[k]\}$ on the grid $\{k\}$, $S^p(k) = z[k],\ k \in \mathbb{Z}$. Application of the direct DFT results in an explicit expression for the spline's coordinates in the orthogonal basis

$$\xi[n] = \hat{z}[n]/u^p[n] \Longrightarrow S^p(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{\hat{z}[n]}{u^p[n]}\,\zeta^p[n](t). \tag{4.31}$$

Remark 4.1. Eq. (4.31) derives the spline's $S^p(t) \in {}^p\mathscr{S}$ coordinates $\{\xi[n]\}$ from its samples at grid points. Conversely, once the coordinates are available, Eq. (4.30) implies that the samples $\{S^p(k)\},\ k = 0, \ldots N-1$ are calculated by the inverse DFT of the sequence $\{\xi[n]\,u^p[n]\}$. The values at the dyadic $\{2^{-r}k\}$ or the triadic $\{3^{-r}k\}$ rational points are derived from the grid samples, using fast subdivision algorithms to be presented in Chap. 6.

Discrete Parseval identity: It follows from Eqs. (1.24) and (4.30)

$$\sum_{k=0}^{N-1} \left|S^p(k)\right|^2 = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} |\xi[n]\,u^p[n]|^2 = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} |\sigma[n]|^2 \frac{(u^p[n])^2}{u^{2p}[n]}. \tag{4.32}$$

Sampled circular convolution: Substituting $t = k$ into Eq. (4.29), we get

$$S^p \circledast S^q(k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n]\,\lambda[n]\,\omega^{kn}\,u^{p+q}[n] \tag{4.33}$$
$$= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \sigma[n]\,\eta[n]\,\omega^{kn}\,Q^{p,q}[n], \quad Q^{p,q}[n] \stackrel{\text{def}}{=} \frac{u^{p+q}[n]}{\sqrt{u^{2p}[n]\,u^{2q}[n]}}.$$

4.1.3.1 S-Filtering

Definition 4.1. The mixed discrete-continuous circular convolution of a spline $S^p(t) \in {}^p\mathscr{S}$ with a signal $\mathbf{h} = \{h[k]\} \in \Pi[N]$

$$G(t) \stackrel{\text{def}}{=} \mathbf{h} \circledast S^p(t) = \sum_{l=0}^{N-1} h[l]\, S^p(t-l) \tag{4.34}$$

is called s-filtering the spline $S^p(t)$.

Proposition 4.2. *Assume the spline $S^p(t) \in {}^p\mathscr{S}$ is represented as in Eq. (4.17) and $\mathbf{h} = \{h[k]\} \in \Pi[N]$. Then the function $G(t) \stackrel{\text{def}}{=} \mathbf{h} \circledast S^p(t)$ is a spline from ${}^p\mathscr{S}$ and*

$$G(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \hat{h}[n]\, \zeta[n](t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \hat{h}[n]\, \gamma[n](t). \tag{4.35}$$

Proof The mixed circular convolution

$$G(t) = \sum_{l=0}^{N-1} h[l]\, S^p(t-l) = \frac{1}{N} \sum_{l=0}^{N-1} h[l] \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta[n](t-l)$$

$$= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta[n](t) \sum_{l=0}^{N-1} h[l]\, \omega^{-nl}\, h[l] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \hat{h}[n]\, \zeta[n](t)$$

$$\frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \hat{h}[n]\, \gamma[n](t).$$

■

Thus, s-filtering a spline results in multiplication of its SHA spectrum with the DFT $\left\{\hat{h}[n]\right\}$ of the signal $\mathbf{h}$. The inverse claims is true.

Proposition 4.3. *Assume the spline $G^p(t) \in {}^p\mathscr{S}$ is represented as*

$$G^p(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \chi[n]\, \zeta[n](t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \chi[n]\, \gamma[n](t). \tag{4.36}$$

Then, it can be represented as a mixed circular convolution $G^p(t) = \sum_{l=0}^{N-1} h[l]$ $S^p(t-l)$, where

$$S^p(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\ \zeta[n](t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \gamma[n](t),$$

$$h[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nl}\, \chi[n].$$

Proof The sequence $\left\{\chi[n] = \hat{h}[n]\right\}$. Then, using the shift property Eq. (4.11), we get

$$\begin{aligned} G^p(t) &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n] \sum_{l=0}^{N-1} \omega^{-nl}\, h[l]\, \zeta[n](t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n] \sum_{l=0}^{N-1} h[l]\, \zeta[n](t-l) \\ &= \sum_{l=0}^{N-1} h[l]\, S^p(t-l), \quad S^p(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\ \zeta[n](t). \end{aligned}$$

■

We call the N-periodic sequence $\left\{\hat{h}[n]\right\}$ the frequency response of the s-filter **h**. If $|\hat{h}[n] > a > 0|$ when $|n| < N/4$ and $\hat{h}[n] \approx 0$ when $|n| > N/4$ then s-filtering retains the low-frequency components of the spline and suppresses the high-frequency components (low-pass s-filtering), and vice versa for high-pass s-filtering.

4.1.4 Generators of Periodic Splines Spaces

An N-periodic spline $\phi(t) \in {}^p\mathscr{S}$ is a generator of the N-periodic splines space ${}^p\mathscr{S}$ if any spline from $S \in {}^p\mathscr{S}$ can be represented as the linear combination

$$S(t) = \sum_{k=0}^{N-1} r[k]\, \phi\,(t-k)\,. \tag{4.37}$$

An obvious generator of ${}^p\mathscr{S}$ is the periodic B-spline $B^p(t)$. Any spline, including candidates for being generators, is represented as

$$\begin{aligned} \phi(t) &= \sum_{k=0}^{N-1} f[k]\, B^p(t-k) \\ &= \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \tau[n] \gamma^p[n](t), \quad \tau[n] = \sqrt{N\, u^{2p}[n]}\, \hat{f}[n]. \end{aligned} \tag{4.38}$$

Recall that the coordinates $\{\tau[n]\}$ of a spline $\phi(t)$ constitute its SHA spectrum. Equation (4.20) implies that

$$\phi^p(t-k) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \tau[n]\gamma^p[n](t-k) \tag{4.39}$$

$$= \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \omega^{-kn}\, \tau[n]\gamma^p[n](t),$$

$$\Longrightarrow \tau[n]\gamma^p[n](t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{kn}\, \phi_d^p(t-k). \tag{4.40}$$

The following describes the condition for a spline to be a generator.

Proposition 4.4. *A spline* $\phi(t) \in {}^p\mathscr{S}$ *represented by* Eq. (4.38) *can serve as a generator of the space* ${}^p\mathscr{S}$ *if its SHA spectrum* $\{\tau[n]\}$ *does not vanish.*

Proof Let a spline $S(t) \in {}^p\mathscr{S}$ be represented as

$$S(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\gamma^p[n](t).$$

Then, using Eq. (4.40), we can write

$$S(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \frac{\sigma[n]}{\tau[n]}\, \tau[n]\, \gamma^p[n](t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \frac{\sigma[n]}{\tau[n]} \sum_{k=0}^{N-1} \omega^{kn}\, \phi^p(t-k)$$

$$= \sum_{k=0}^{N-1} r[k]\, \phi^p(t-k), \quad r[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn} \frac{\sigma[n]}{\tau[n]}. \tag{4.41}$$

Proposition 4.5. *Assume a spline* $\phi^p(t)$ *presented as in* Eq. (4.38) *is a generator of* ${}^p\mathscr{S}$. *Then there exists the unique dual generator*

$$\phi_d^p(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \theta[n]\gamma^p[n](t)$$

such that the the biorthogonal relations

$$\int_0^N \phi^p(t-k)\, \left(\phi_d^p(t-m)\right)^*\, dt = \delta[k-m] \tag{4.42}$$

hold. The SHA spectra $\{\tau[n]\}$ *and* $\{\theta[n]\}$ *of the generators* $\phi^p(t)$ *and* $\phi_d^p(t)$, *respectively, are linked as*

$$\tau[n]\, \theta[n]^* = 1. \tag{4.43}$$

Proof Equation (4.39) implies that the splines $\phi^p(t-k)$ and $\phi_d^p(t-m)$ are

$$\phi^p(t-k) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \omega^{-kn}\, \tau[n]\gamma^p[n](t),$$

$$\phi_d^p(t-m) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \omega^{-kn}\, \theta[n]\gamma^p[n](t).$$

Then, using the Parseval identity (4.25), and the relation (4.43), we get

$$\int_0^N \phi^p(t-k)\, \phi_d^p(t-m)^*\, dt = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \tau[n]\, \theta[n]^* \omega^{-n(k-m)} = \delta[k-m].$$

■

Remark 4.1.2. If a spline is expanded over the shifts of a generator $\phi^p(t)$ as $S(t) = \sum_{k=0}^{N-1} r[k]\, \phi^p\,(t-k)$ and $\phi_d^p(t)$ is the dual generator then

$$\left\langle S, \phi_d^p(\cdot - m)\right\rangle = \int_0^N \sum_{k=0}^{N-1} r[k]\, \phi^p(t-k)\, \phi_d^p(t-m)^*\, dt = r[m]. \tag{4.44}$$

4.1.4.1 Examples of Generators of the Periodic Splines Spaces

extbfPeriodic B-spline: Due to Eq. (3.18),

$$B^p(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \zeta^p[n](t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sqrt{u^{2p}[n]}\gamma^p[n](t). \tag{4.45}$$

Thus, the SHA spectrum of the B-spline $B^p(t)$ is $\left\{\sqrt{u^{2p}[n]}\right\}$, $n = -N/2, \ldots N/2-1$. Figure 4.1 displays the B-splines of orders 2, 4, 8 and 12 and their SHA spectra. Visually, the SHA spectrum of twelfth order B-spline is similar to the Gaussian curve.

Generator dual to periodic B-spline: Due to Eq. (4.43), the spline

$$D^p(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \frac{\gamma^p[n](t)}{\sqrt{u^{2p}[n]}} \tag{4.46}$$

is dual to the B-spline $B^p(t)$. The SHA spectrum of $D^p(t)$ is $\left\{1/\sqrt{u^{2p}[n]}\right\}$, $n = -N/2, \ldots N/2-1$.

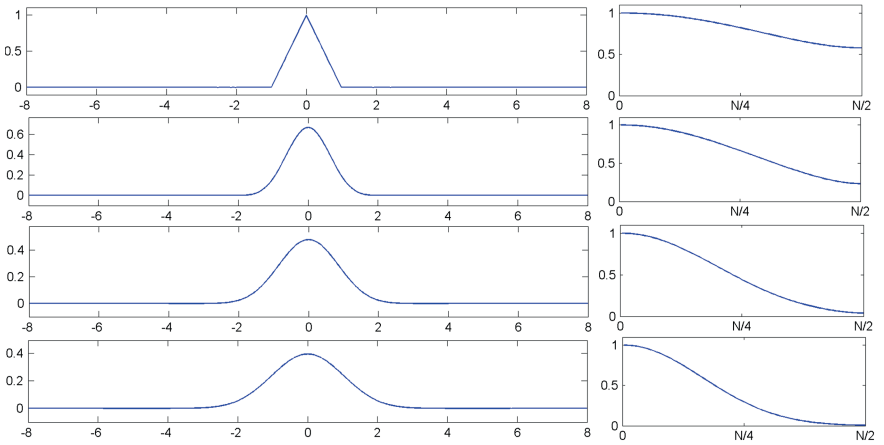


Fig. 4.1 $N-$periodic B-splines $B^p(t)$ of orders 2, 4, 8 and 12 and their SHA spectra, $N = 256$

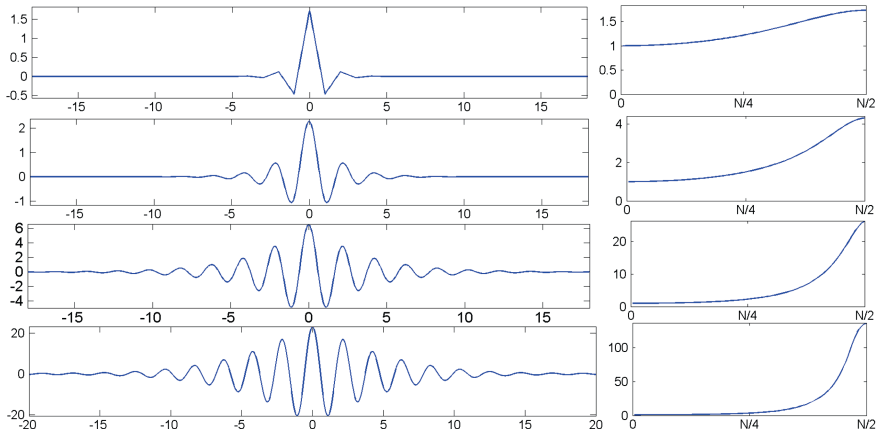


Fig. 4.2 $N-$periodic DB-splines $D^p(t)$ dual to B-splines of orders 2, 4, 8, and 12 and their SHA spectra, $N = 256$

Figure 4.2 displays the dual splines $D^p(t)$ of orders 2, 4, 8 and 12 and their SHA spectra.

The interpolating generator (fundamental periodic spline): The so-called $N-$periodic fundamental spline is

$$L^p(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \frac{\zeta^p[n](t)}{u^p[n]} = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \frac{\gamma^p[n](t)}{V^p[n]}, \quad V^p[n] = \frac{u^p[n]}{\sqrt{u^{2p}[n]}}. \tag{4.47}$$

The B-spline representation of the spline $L^p(t)$ is

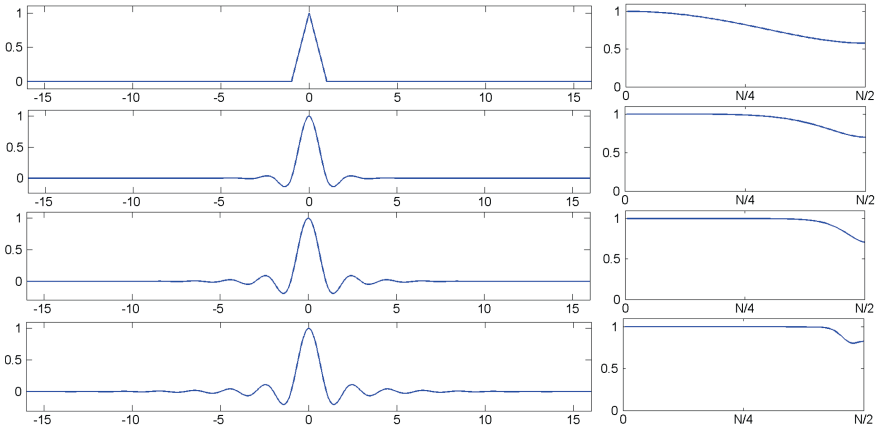


Fig. 4.3 $N-$periodic interpolating generators $L^p(t)$ of orders 2, 4, 8, and 12 and their SHA spectra, $N = 256$

$$L^p(t) = \sum_{k=0}^{N-1} \lambda[k]\, B^p(t-k), \quad \hat{\lambda}[n] = \frac{1}{u^p[n]}. \tag{4.48}$$

It follows from Eq. (4.30) that the values of the spline at grid points

$$L^p(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn} = \delta[k](\mathrm{mod} N). \tag{4.49}$$

The fundamental spline provides an explicit representation for the spline $S(t) \in {}^p\mathscr{S}$, which interpolates a signal $\mathbf{z} = \{z[k]\}$ at grid points:

$$S(t) = \sum_{k=0}^{N-1} z[k]\, L^p(t-k), \quad S(n) = \sum_{k=0}^{N-1} z[k]\, L^p(n-k) = z[n]. \tag{4.50}$$

Figure 4.3 displays the interpolating generators $L^p(t)$ of orders 2, 4, 8 and 12 and their SHA spectra.

Generator dual to the periodic fundamental spline: The spline

$$M^p(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} V^p[n]\, \gamma^p[n](t) \tag{4.51}$$

is dual to the fundamental spline $L^p(t)$. Figure 4.4 displays the generators $M^p(t)$ of orders 2, 4, 8 and 12 and their SHA spectra.

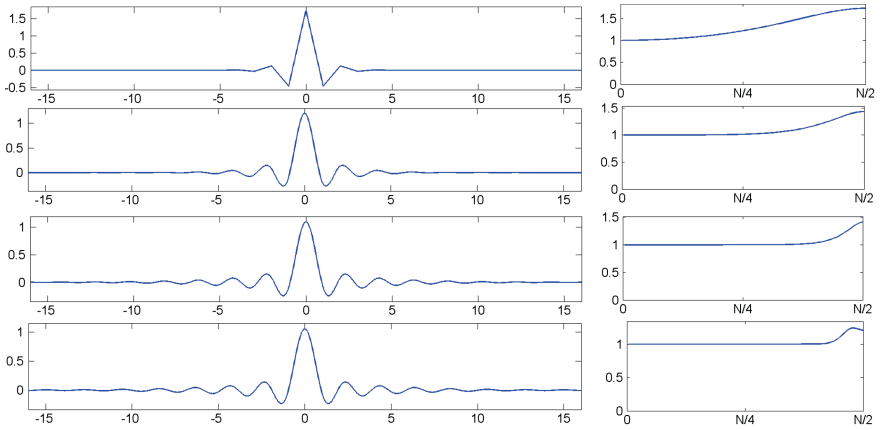


Fig. 4.4 $N-$periodic generators $M^p(t)$ dual to interpolating ones of orders 2, 4, 8, and 12 and their SHA spectra, $N = 256$

The generator whose shifts are orthogonal to each other (self-dual generator): It is readily seen from Eq. (4.43) that the SHA spectrum of such a spline should be $\tau[n] = 1,\ n = -N/2, ...N/2 - 1$. Thus,

$$\varphi^p(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \gamma^p[n](t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \frac{\zeta^p[n](t)}{\sqrt{u^{2p}[n]}}, \quad \left\|\varphi^p(t)\right\| = 1. \quad (4.52)$$

Consequently, the splines $\{\varphi^p(t-k)\}$, $k = 0, ...N-1$, form an orthonormal basis of the space ${}^p\mathscr{S}$. Any spline $S(t) \in {}^p\mathscr{S}$ can be expanded as

$$S(t) = \sum_{k=0}^{N-1} s[k]\,\varphi^p(t-k), \quad s[k] = \int_0^N S(t)\,\varphi^p(t-k)\,dt. \quad (4.53)$$

The splines $\varphi^p(t)$ are the periodizations of the so-called Battle–Lemarié father wavelets [2, 3].

Remark 4.1.3. The SHA spectra of all the above generators are positive and symmetric about $n = 0$. It follows from properties of the sequences $u^p[n]$.

Graphical illustrations Figures 4.1, 4.2, 4.3, 4.4 and 4.5 display the generators of the splines spaces ${}^p\mathscr{S}$ where the orders $p = 2, 4, 8, 12$. In addition, Figs. 4.1, 4.2, 4.3 and 4.4 display the SHA spectra (right hand half-band) of all the generators except for the self-dual ones, whose SHA spectra are identically equal to 1. All the figures were produced by the MATLAB code `GensplConsP.m`. This code is utilizing the binary spline subdivision scheme to be described in Sect. 6.2. Examples of generators of odd-order splines spaces are displayed in Figs. 6.1 and 6.2 after introduction of the

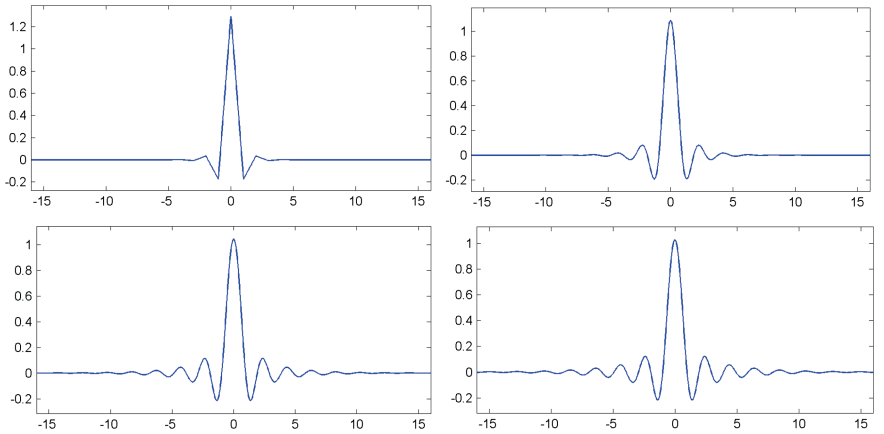


Fig. 4.5 $N-$periodic self-dual generators $\varphi^p(t)$ of orders 2, 4, 8, and 12, $N = 256$. *Top left* $p = 2$. *Top right* $p = 4$. *Bottom left* $p = 8$. *Bottom right* $p = 12$

ternary spline subdivision scheme. The structure of Figs. 4.1, 4.2, 4.3 and 4.4 is the following: The left panels display the generators while the right panels display their SHA spectra. Top pair: $p = 2$, second from top pair: $p = 4$, second from bottom pair: $p = 8$, bottom pair: $p = 12$.

Comment

We observe that, as the spline's order grows, the SHA spectra of the fundamental splines and their dual counterparts tend to 1. It stems from the fact that the sequence $V^p[n] = u^p[n]/\sqrt{u^{2p}[n]} \to 1$ as $p \to \infty$ Visually, the shapes of the fundamental spline of twelfth order and of its dual counterpart are very similar to each other and similar to the shape of the self-dual generator.

4.1.4.2 Fundamental Splines and the Sinc Function

Equation (4.49) claims that the grid samples of the fundamental spline $\varphi^p(k) = \delta[k](\mathrm{mod}N)$. This property is similar to the property of the *sinc* function $\mathrm{sinc}(t) \stackrel{\mathrm{def}}{=} \sin \pi t/\pi t$. The *sinc* function represents bandlimited functions via their samples ([6], for example): If the Fourier spectrum of a non-periodic signal $f(t)$ is located within the band $[-\sigma, \sigma]$, then it can be represented as $f(t) = \sum_{k\in\mathbb{Z}} f(Pk)\,\mathrm{sinc}(t/P - k)$, where $P = \pi/\sigma$.

The Fourier transform of the *sinc* function is $C(v) \stackrel{\mathrm{def}}{=} \int_{-\infty}^{\infty} e^{-ivt}\,\mathrm{sinc}(t)\,dt = \chi[-\pi, \pi](v)$, where the function χ is defined in Eq. (3.12). Denote by $\mathrm{Sinc}_N(t) \stackrel{\mathrm{def}}{=}$

$\sum_{l\in\mathbb{Z}} \mathrm{sinc}\,(t + lN)$ the periodized *sinc* function. The Fourier coefficients of the periodic function $\mathrm{Sinc}_N\,(t)$ are

$$c_n\,(\mathrm{Sinc}_N) = C\left(\frac{2\pi n}{N}\right) = \begin{cases} 1, \text{ if } |n| \le N/2; \\ 0, \text{ otherwise} \end{cases}$$
$$\Longrightarrow \mathrm{Sinc}_N(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2} e^{2\pi i n t/N} = \frac{\sin \pi t(1+1/N)}{N\,\sin \pi t/N}.$$

Apparently, $\mathrm{Sinc}_N(t) \to \mathrm{sinc}(t)$ as $t \in -N/2, N/2)$ and the period $N \to \infty$.

The Fourier spectra of the fundamental splines $\varphi^p(t)$ approach the Fourier spectrum of the *sinc* function when the splines orders increase. Such a convergence is established in [1] for the non-periodic fundamental splines. The statement for the periodic case is the following.

Theorem 4.1. *All the Fourier coefficients $c_n(\varphi^p)$, where $|n| < N/2$, of the $N-$periodic fundamental splines tend to 1 when the spline order p is growing. The coefficients, whose indices $n > N/2$ or $n < -N/2$ tend to zero as p is growing. The coefficients $c_{\pm N/2}(\varphi^p)$ tend to 1/2 as p is growing.*

Proof We prove the statement for the even-order splines $\varphi^{2r}(t)$. Proof for the odd order splines differs in a few non-essential details. Equations (3.16), (4.5) and (4.48) imply that the Fourier coefficients are

$$c_n[\varphi^{2r}] = \frac{1}{u^{2r}[n]}\left(\frac{\sin \pi n/N}{\pi n/N}\right)^{2r} = \frac{1}{D^{2r}[n]},$$
$$D^{2r}[n] \stackrel{\text{def}}{=} \left(\frac{\pi n}{N}\right)^{2r}\sum_{l\in\mathbb{Z}}\frac{1}{(\pi\,(n/N+l))^{2r}} = \sum_{l\in\mathbb{Z}}\frac{1}{(1+N\,l/n)^{2r}} = 1+\sigma[n,2r],$$

where the sequence

$$\sigma[n,2r] \stackrel{\text{def}}{=} \sigma_+[n,2r]+\sigma_-[n,2r], \quad \sigma_\pm[n,2r] \stackrel{\text{def}}{=} \sum_{l=1}^{\infty}\frac{1}{(1\pm N\,l/n)^{2r}}.$$

Note that the coefficients $c_{lN} = 0$ for all the integers $l \neq 0$, while $c_0 = 1$ and all the coefficients are non-negative and do not exceed 1. The series $\sigma_\pm[n, 2r]$ converge with any integer $n \neq kN$. Assume that $0 < n < N/2$. The terms in the series $\sigma_+[n, 2r]$ decrease monotonically as l grows. Therefore, the following inequalities are true

$$\sigma_+[n,2r] < \frac{1}{(1+N/n)^{2r}} + \int_1^{\infty}\frac{1}{(1+Nt/n)^{2r}}\,dt < \alpha_+[2r],$$
$$\alpha_+[2r] \stackrel{\text{def}}{=} \frac{1}{3^{2r}} + \frac{2r+1}{2^{2r+2}}.$$

The sum of the series $\sigma_-[n, 2r]$ is

$$\sigma_-[n, 2r] < \frac{1}{(1 - N/n)^{2r}} + \frac{1}{(1 - 2N/n)^{2r}} + \int_2^\infty \frac{1}{(1 - Nt/n)^{2r}}\, dt < \alpha_-[n, 2r],$$

$$\alpha_-[n, 2r] \stackrel{\text{def}}{=} \left(\frac{N - n}{n}\right)^{2r} + \alpha_+[2r].$$

Consequently,

$$c_n[\varphi^{2r}] = \frac{1}{1 + \sigma[n, 2r]} = 1 - \sigma[n, 2r] + \sum_{l=2}^{\infty} (-\sigma[n, 2r])^l = 1 - \beta[n, 2r],$$

$$0 < \beta[n, 2r] < \sigma[n, 2r] < \left(\frac{N - n}{n}\right)^{2r} + 2\,\alpha_+[2r].$$

Thus, each coefficient $c_n[\varphi^{2r}],\ 0 \le n < N/2$, is approaching 1 as the spline order $2r$ increases. Recall that the order $2r$ cannot exceed $N - 1$. The convergence speed is different for different n. When n is approaching $N/2$, then the convergence speed is decreasing. Obviously, the same claim is true for $-N/2 < n < 0$.

If $n > N/2$ then $1 - N/n < 1$. Thus, the first term in the series $\sigma_-[n, 2r]$, which is $(1 - N/n)^{-2r}$, grows when $2r$ increases, while the sum of the remainder of the series is reduced as well as the series $\sigma_+[n, 2r]$. As a result, the coefficient $c_n([\varphi^{2r}] = 1/D^{2r}[n] = O\left((1 - N/n)^{2r}\right)$. When $n > N/2$ is growing then the coefficients are converging fast to 0.

Let $n = N/2$. Then,

$$\sigma_+[n, 2r] = \sum_{l=1}^{\infty} \frac{1}{(1 + 2l)^{2r}} < \frac{1}{2^{2r}}\, C_+, \quad C_+ \stackrel{\text{def}}{=} \sum_{l=1}^{\infty} \frac{1}{l^{2r}},$$

$$\sigma_-[n, 2r] = 1 + \beta_-[2r],\ \beta_-[2r] \stackrel{\text{def}}{=} \sum_{l=2}^{\infty} \frac{1}{(1 - 2l)^{2r}} < \frac{1}{2^{2r}}\, C_-,\ C_- \stackrel{\text{def}}{=} \sum_{l=2}^{\infty} \frac{1}{l^{2r}}.$$

Thus, $D^{2r}[\pm N/2] = 2 + O(2^{-2r})$. ■

Corollary 4.1. *The fundamental splines $\varphi^p(t)$ are approaching the periodic sinc function* $\mathrm{Sinc}(t)$ *as the spline orders p are increasing.*

Figure 4.6 displays the fundamental splines $\varphi^p(t)$ of orders 2, 6, 10, 14 and their Fourier coefficients. We observe that the effective supports of the splines widen and their behavior becomes oscillating as the orders increase, while the shapes of their spectra tend to the rectangle.

Figure 4.7, which was produced by the MATLAB code `Sinc_examp.m`, compares the fundamental spline $\varphi^{22}(t)$ of order 22 with the periodic *sinc* function $\mathrm{Sinc}_N(t)$, $N = 512$. When t is not far from zero (up to 5), the two functions

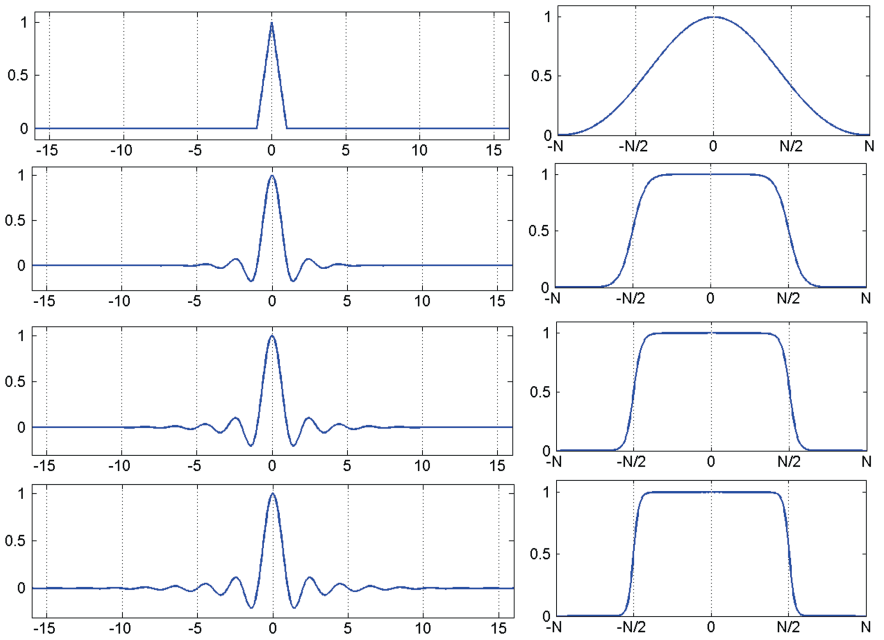


Fig. 4.6 *Left* fundamental splines $\varphi^{2r}(t)$ of orders (from *top* to *bottom*) 2, 6, 10, 14. *Right* their Fourier coefficients

are almost undistinguishable. When t grows, the spline practically vanishes while $\mathrm{Sinc}_N(t)$ oscillates. The spectrum of the spline is near a perfect rectangle.

4.1.5 Remarks on Orthonormal Bases of Splines Spaces and SHA Spectra

So far, we have introduced two alternative orthonormal bases of the periodic splines space ${}^p\mathscr{S}$: one basis – $\Gamma^p \stackrel{\text{def}}{=} \{\gamma^p[n](t)\}$, $n = -N/2, ..., N/2-1$, consisting of normalized periodic exponential splines, which were defined in Eq. (4.16), and the other basis – $\Phi^p \stackrel{\text{def}}{=} \{\varphi^p(t-k)\}$, $k = 0, ..., N-1$, consisting of shifts of the self-dual generator $\varphi^p(t)$, which was defined in Eq. (4.52). These bases are linked to each other via the direct and inverse DFT:

$$\varphi^p(t-k) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \omega^{-nk} \gamma^p[n](t), \quad \gamma^p[n](t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{nk} \varphi^p(t-k). \tag{4.54}$$

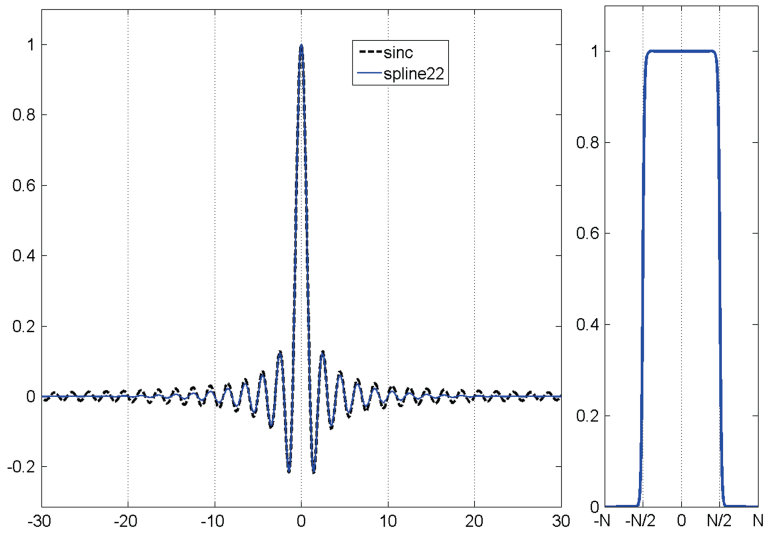


Fig. 4.7 *Left* fundamental splines $\varphi^{22}(t)$ of order 22 (*solid line*) and *sinc* function $\mathrm{Sinc}_N(t)$, $N = 512$ (*dashed line*). *Right* the Fourier coefficients of this spline

The exponential splines $\gamma^p[n](t)$ are almost perfectly localized in the frequency domain. Fourier expansion of $\gamma^p[n](t)$ stems from Eq. (4.2). For simplicity, assume that the splines orders are even, $p = 2r$. Then,

$$\gamma^{2r}[n](t) = \frac{\zeta^{2r}[n](t)}{\sqrt{N\, u^{4r}[n]}} = \frac{\sin^{2r} \pi n/N}{\sqrt{N\, u^{4r}[n]}} \sum_{l\in\mathbb{Z}} \frac{e^{2\pi i (n/N+l)t}}{(\pi\,(n/N+l))^{2r}} \tag{4.55}$$

$$= \frac{1}{\sqrt{N\, u^{4r}[n]}} \left(\frac{\sin \pi n/N}{\pi n/N}\right)^{2r} \sum_{l\in\mathbb{Z}} \frac{e^{2\pi i (n/N+l)t}}{(1+Nl/n)^{2r}}.$$

Equation (4.5) implies that

$$\sqrt{u^{4r}[n]} = \sin^{2r} \pi n/N \sqrt{\sum_{m\in\mathbb{Z}} \left(\frac{1}{\pi\,(n/N+m)}\right)^{4r}} \tag{4.56}$$

$$= \left(\frac{\sin \pi n/N}{\pi n/N}\right)^{2r} \left(1 + \sum_{m\in\mathbb{Z}\setminus 0} \left(\frac{1}{1+Nm/n}\right)^{4r}\right)^{1/2}$$

$$= \left(\frac{\sin \pi n/N}{\pi n/N}\right)^{2r} \left(1 + O(N^{-2r})\right).$$

Substitution of Eq. (4.56) into Eq. (4.55) results in the relation

$$\gamma^{2r}[n](t) = \frac{1}{\sqrt{N}}\left(e^{2\pi int/N} + \sum_{l\in\mathbb{Z}\setminus 0}\frac{e^{2\pi i(n/N+l)t}}{(1+Nl/n)^{2r}} + O(N^{-2r})\right),\ n = -\frac{N}{2},..\frac{N}{2}-1.$$

Thus, the spline of even order $\gamma^{2r}[n](t)$ is a linear combination of the exponential function $e^{2\pi int/N}/\sqrt{N}$ and its overtones, which decay as $(Nl)^{-2r}$. This claim is true for the splines of odd orders $p = 2r+1$ as well. The complex-valued splines $\gamma^p[n](t)$ provide a substitute of the Fourier harmonics for the periodic spline spaces ${}^p\mathscr{S}$. Their properties are similar to the properties of the Fourier harmonics, which were listed in Sect. 1.1.1. Note that, like the splines $\gamma^p[n](t)$, the functions $e^{2\pi int/N}/\sqrt{N}$ are orthogonal to each other and their norms are equal to one.

Assume a spline $S(t) \in {}^p\mathscr{S}$ is represented as

$$S(t) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\sigma[n]\,\gamma^p[n](t).$$

The set of coefficients $\{\sigma[n]\},\ n = -N/2, ...N/2-1$, completely determines the spline $S(t)$. These coefficients have an obvious spectral meaning. Due to orthonormality of $\gamma^p[n](t)$,

$$\begin{aligned}\sigma[n] &= \sqrt{N}\int_0^N S(t)\,\gamma^p[n](t)^*\,dt\\ &= \int_0^N S(t)\,e^{-2\pi int/N}\,dt + O(N^{-p}) = c_n(S) + O(N^{-p}).\end{aligned}$$

Thus, the SHA spectrum components $\{\sigma[n]\},\ n = -N/2, ...N/2-1$, of a spline are close to its Fourier coefficients $\{c_n(S)\},\ n = -N/2, ...N/2-1,$. Note that the Fourier coefficients $\{c_n(S)\}$, whose indices n are outside the range $-N/2, ...N/2-1$, are decaying as n^{-p}. Therefore the SHA spectrum characterizes the splines behavior in the frequency domain.

The complex-valued splines $\gamma^p[n](t)$ are not localized in time domain. On the contrary, the generator $\varphi^p(t)$, whose shifts form an orthonormal basis of ${}^p\mathscr{S}$ is real-valued and well localized in time domain, as it is seen in Fig. 4.5. The expansion coefficients of a spline over the basis $\{\varphi^p(t-k)\}$ characterizes the splines behavior in the time domain. The generators $\varphi^p(t)$ are not localized in the Fourier domain and their SHA spectra are all ones. Due to the shift property (4.20) of the exponential splines,

$$\varphi^p(t-k) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\omega^{-nk}\,\gamma^p[n](t). \tag{4.57}$$

As for the non-normalized exponential splines $\{\zeta^p[n](t)\}$, which form an orthogonal basis of ${}^p\mathscr{S}$, they are related to the non-normalized Fourier exponentials $e^{2\pi int/N}$. Recall the interpolation relation Eq. (4.13). Similarly to the spline $\gamma^p[n](t)$, the spline $\zeta^p[n](t)$ is a linear combination of the exponential function $e^{2\pi int/N}$ and its overtones, which decay as $(Nl)^{-p}$. The time domain counterpart of the orthogonal basis $\{\zeta^p[n](t)\}$ of ${}^p\mathscr{S}$ is the basis consisting of the shifts $\{B^p(t-k)\}$ of the B-splines.

4.2 SHA in Two-Dimensional Spline Spaces

For simplicity, assume that the two-dimensional splines to be discussed have the same periodicity in x and y directions. The 2D N-periodic B-spline on the grid $\{k, n\}$ is defined as a product of 1D B-splines $B^{p,q}(x, y) \stackrel{\text{def}}{=} B^p(x) \cdot B^q(y)$. Similarly, the N-periodic 2D exponential splines $\zeta^{p,q}[\kappa, \nu](x, y) \stackrel{\text{def}}{=} \zeta^p[\kappa](x) \cdot \zeta^q[\nu](y)$ and $\gamma^{p,q}[\kappa, \nu](x, y) \stackrel{\text{def}}{=} \gamma^p[\kappa](x) \cdot \gamma^q[\nu](y)$ are defined. 2D splines are defined as linear combinations of the 2D B-splines

$$S^{p,q}(x, y) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} s[k, n]\, B^p(x-k)\, B^q(y-n). \tag{4.58}$$

Note that, once the variable y is fixed, $S^{p,q}(x, y)$ becomes a 1D spline of order p with respect to x, and vice versa for fixed x. The space of such splines is denoted as ${}^{p,q}\mathscr{S}$. Alternatively,

$$\begin{aligned} S^{p,q}(x, y) &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa, \nu]\, \zeta^{p,q}[\kappa, \nu](x, y), \\ &= \frac{1}{N} \sum_{\kappa,\nu=-N/2}^{N/2-1} \sigma[\kappa, \nu]\, \gamma^{p,q}[\kappa, \nu](x, y), \\ \xi[\kappa, \nu] = \hat{s}[\kappa, \nu], &\quad \sigma[\kappa, \nu] = \xi[\kappa, \nu] \sqrt{u^{2p,2q}[\kappa, \nu]}, \end{aligned} \tag{4.59}$$

where

$$u^{p,q}[\kappa, \nu] \stackrel{\text{def}}{=} u^p[\kappa] \cdot u^q[\nu] \tag{4.60}$$

denotes the characteristic function of the space ${}^{p,q}\mathscr{S}$ and $\hat{s}[\kappa, \nu]$ denotes the 2D DFT of the coefficients $s[k, n]$. The coefficients $s[k, n]$ are linked to $\xi[\kappa, \nu]$ and $\sigma[\kappa, \nu]$ via the 2D IDFT:

$$s[k,n] = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \omega^{k\kappa+n\nu}\, \xi[\kappa,\nu] = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \omega^{k\kappa+n\nu}\, \frac{\sigma[\kappa,\nu]}{\sqrt{u^{2p,2q}[\kappa,\nu]}}.$$

The splines $\gamma^{p,q}[\kappa,\nu](x,y)$ form an orthonormal basis of ${}^{p,q}\mathscr{S}$, and the set of coordinates $\{\sigma[\kappa,\nu]\}$ of a spline $S^{p,q}(x,y)$ in this basis is called its 2D SHA spectrum. The splines $\zeta^{p,q}[\kappa,\nu](x,y)$ form an orthogonal basis of ${}^{p,q}\mathscr{S}$. Extension of the SHA to 2D spline spaces is straightforward. In particular:

Sampling and interpolation: Let $S^{p,q}(x,y) \in {}^{p,q}\mathscr{S}$ be represented as in Eq. (4.59). Then, Eqs. (4.11) and (4.20) imply that grid samples

$$\begin{aligned} S^{p,q}(k,n) &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa,\nu]\, \zeta[\kappa,\nu]^{p,q}(k,n) \\ &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa,\nu] u^p[\kappa]\, u^q[\nu]\ \omega^{(\kappa k+\nu n)}, \end{aligned} \tag{4.61}$$

If the spline $S^{p,q}$ interpolates the array $\mathbf{z} = \{z[k,n]\}$, $k,n, = 0, ..., N-1$,

$$S^{p,q}(k,n) = z[k,n] \Longleftrightarrow \xi[\kappa,\nu] = \frac{\hat{z}[\kappa,\nu]}{u^p[\kappa]\, u^q[\nu]}. \tag{4.62}$$

Parseval identities:

$$\begin{aligned} \|(S^{p,q})^{(s,t)}_{x,y}\|^2 &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \tilde{W}^{p,s}[\kappa]\ \ \tilde{W}^{q,t}[\nu]\ |\xi[\kappa,\nu]|^2 \\ &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} W^{p,s}[\kappa]\ \ W^{q,t}[\nu]\ |\sigma[\kappa,\nu]|^2, \end{aligned} \tag{4.63}$$

where the sequences are

$$\tilde{W}^{p,s}[n] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi n}{N}\right)^{2s} u^{2(p-s)}[n], \quad W^{p,s}[n] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi n}{N}\right)^{2s} \frac{u^{2(p-s)}[n]}{u^{2p}[n]}.$$

Circular convolution: Let $S^{p,q} \in {}^{p,q}\mathscr{S}$ be represented as in Eq. (4.59) and

$$S^{\bar{p},\bar{q}}(x,y) = \frac{1}{N} \sum_{\kappa,\nu=-N/2}^{N/2-1} \mu[\kappa,\nu]\, \zeta^{\bar{p},\bar{q}}[\kappa,\nu](x,y) \in {}^{\bar{p},\bar{q}}\mathscr{S}. \tag{4.64}$$

Then the circular convolution

$$
\begin{aligned}
& S^{p,q} \circledast S^{\bar{p},\bar{q}}(x, y) \\
&= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa, \nu]\, \mu[\kappa, \nu]\, \zeta^{(p+\bar{p}),(q+\bar{q})}[\kappa, \nu](x, y) \in {}^{(p+\bar{p}),(q+\bar{q})}\mathscr{S}, \\
& S^{p,q} \circledast S^{\bar{p},\bar{q}}(k, n) \\
&= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \omega^{(k\kappa+n\nu)}\, \xi[\kappa, \nu]\, \mu[\kappa, \nu]\, u^{p+\bar{p},q+\bar{q}}[\kappa, \nu].
\end{aligned}
\tag{4.65}
$$

References

1. A. Aldroubi, M. Unser, M. Eden, Cardinal spline filters: stability and convergence to the ideal sinc interpolator. Signal Process. **28**(2), 127–138 (1992)
2. G. Battle, A block spin construction of ondelettes. I. lemarié functions. Commun. Math. Phys. **110**(4), 601–615 (1987)
3. P. G. Lemarié. Ondelettes à localisation exponentielle. J. Math. Pures Appl. (9), **67**(3), 227–236 (1988)
4. P. Neittaanmäki, V. Rivkind, V. Zheludev, *A wavelet transform based on periodic splines and finite element method*, ed. by M. Kvectorrívectorzek, P. Neittaanmäki, R. Stenberg, Finite Element Methods (Jyväskylä, 1993), vol. 164 of Lecture Notes in Pure and Appl. Math. (New York, Dekker, 1994), pp. 325–334
5. P. Neittaanmäki, V. Rivkind, V. Zheludev, Periodic spline wavelets and representation of integral operators. Preprint 177, University of Jyväskylä, Department of Mathematics (1995)
6. C.E. Shannon, W. Weaver, *The Mathematical Theory of Communication* (The University of Illinois Press, Urbana, IL, 1949)
7. V. Zheludev, An operational calculus connected with periodic splines. Soviet. Math. Dokl. **42**(1), 162–167 (1991)
8. V. Zheludev, Periodic splines and the fast Fourier transform. Comput. Math. Math. Phys. **32**(2), 149–165 (1992)
9. V. Zheludev, *Periodic splines, harmonic analysis, and wavelets*, ed. by Y. Y. Zeevi, R. Coifman, Signal and Image Representation in Combined Spaces, vol. 7 of Wavelet Anal. Appl. (Academic Press, San Diego, CA, 1998), pp. 477–509

Chapter 5
Polynomial Smoothing Splines

Abstract Interpolating splines is a perfect tool for approximation of a continuous-time signal $f(t)$ in the case when samples $x[k] = f(k),\ k \in \mathbb{Z}$ are available. However, frequently, the samples are corrupted by random noise. In such case, the so-called smoothing splines provide better approximation. In this chapter we describe periodic smoothing splines in one and two dimensions. The SHA technique provides explicit expression of such splines and enables us to derive optimal values of the regularization parameters.

When available samples of a signal $f(t)$ are corrupted by random noise, it should be approximated from the data $\mathbf{y} = \{y[k] = f(k) + \varepsilon_k\},\ k \in \mathbb{Z}$, where $\mathbf{e} = \{\varepsilon_k\},\ k \in \mathbb{Z}$, is a vector of random errors. Exact interpolation of the data y does not make sense because it will result in an irregular function, which hardly fits the signal $f(t)$, especially if $f(t)$ is smooth. A natural idea[1] is to relax the interpolation requirement while introducing a smoothness constraint. This approach was formalized by Schoenberg in his 1964 paper [2] and Reinsch [3].

5.1 One-Dimensional Smoothing Splines

Minimal norm property of even-order splines

Splines of even orders from the spaces ${}^{2r}\mathscr{S}$ possess a remarkable property. Assume $\mathbf{x} = \{x[k]\}$ is an N-periodic signal and denote by $\mathscr{F}_x^r$ the space of N-periodic functions $f(t)$ such that the functional $I(f) \stackrel{\text{def}}{=} \int_0^N (f^{(r)}(t))^2\,dt < \infty$ and $f(k) = x[k],\ k \in \mathbb{Z}$.

[1] It can be traced to as far back as the 1923 Whittaker paper [1]

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_5,

Proposition 5.1 ([4, 5]) *The spline* $S^{2r}(t) \in {}^{2r}\mathscr{S}$ *of even order* $2r$*, which interpolates the signal* $\mathbf{x}$*, yields the minimum to the functional* $I(f)$ *on the space* $\mathscr{F}_x^r$*:*

$$S^{2r}(t) = \arg \min_{f \in \mathscr{F}_x^r} \int_0^N (f^{(r)}(t))^2 \, dt. \tag{5.1}$$

In particular, the cubic interpolatory spline $S^4(t)$ *minimizes the "energy" integral*

$$S^4(t) = \arg \min_{f \in \mathscr{F}_x^2} \int_0^N (f''(t))^2 \, dt. \tag{5.2}$$

The claim of the proposition remains true even when the signals samples are defined on a non-uniform grid. The integral in Eq. (5.2) is related to the curvature, therefore the cubic spline's plot describes a minimum curvature line, which passes through a given set of points on the plane. To draw such lines, draftsmen used splines that were made flexible wood or metal strips. Mathematical splines owe their name to that fact.

Minimization problems for noised data

Denote by $\mathscr{F}^r$ the subspace of the continuous-time $N-$periodic signals $\varphi(t)$ such that the functional $I(\varphi) \stackrel{\text{def}}{=} \int_0^T (\varphi^{(r)}(t))^2 \, dt < \infty$. Assume that the signal to be approximated is $f(t) \in \mathscr{F}^r$, and the available discrete data is $\mathbf{y} = \{y[k] = f(k) + e_k\}$, $k = 0, \ldots, N-1$, where $\mathbf{e} = \{e_k\}$ is the vector of random zero-mean independent identically distributed (i.i.d.) errors, which is a white noise. In addition, assume that the sum $\varepsilon^2 \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} e_k^2$ can be evaluated.

It is natural to approximate the signal $f(t)$ by a function $g(t) \in \mathscr{F}^r$, which yields the minimum to the functional I subject to the condition that the discrepancy functional $E_y(g) \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} |g(k) - y[k]|^2 \le \varepsilon^2$.

This constrained minimization problem is reduced to the solution of the unconstrained problem:

$$g_\rho(t) = \arg \min_{g \in \mathscr{F}^r} (J_\rho(g)), \quad J_\rho(g) \stackrel{\text{def}}{=} \rho \, I(g) + E(g),$$

which is followed by derivation of the numerical parameter ρ from the equation $e(\rho) \stackrel{\text{def}}{=} E_y(g_\rho) = \varepsilon^2$.

5.1.1 Solution of the Unconstrained Minimization Problem

Proposition 5.2 [2] *A unique solution to the minimization problem* $\min_{g \in \mathscr{F}^r} J_\rho(g)$ *is a spline* $S_\rho[\mathbf{y}](t) \in {}^{2r}\mathscr{S}$ *of even order* $2r$.

Proof Assume function $g(t)$ from the space $\mathscr{F}^r$ brings the functional $J_\rho(g)$ to the minimum. Define by $S_\rho[\mathbf{y}](t) \in {}^{2r}\mathscr{S}$ a spline, which interpolates the function $g(t)$ at grid points. Since $S_\rho[\mathbf{y}](k) = g(k)$, the functional $E_y(S_\rho) = E_y(g)$. However, due to the minimal norm property (Proposition 5.1), the functional $I(S_\rho) \le I(g)$. Consequently, $J_\rho(S_\rho) \le J_\rho(g)$. Therefore, the function $g(t) \equiv S_\rho[\mathbf{y}](t)$. ■

The spline $S_\rho^{2r}(t) \in {}^{2r}\mathscr{S}$, which minimizes the functional J_ρ, is called the periodic smoothing spline. The spline $S_\rho(t)$ can be explicitly expressed via the exponential splines basis.

Assume a spline $S(t) \in {}^{2r}\mathscr{S}$ is represented as

$$S^{2r}(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta^{2r}[n](t).$$

Equation (4.27) implies that the functional

$$I(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} w^{2r}[n]\, u^{2r}[n]\, |\xi[n]|^2, \quad w^{2r}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^{2r}. \tag{5.3}$$

Due to Eq. (4.30), the grid samples of $S^{2r}(t)$ are $S^{2r}(k) = N^{-1} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \omega^{kn}\, u^{2r}[n]$. In turn, the signal $\mathbf{y}$ is expressed via its DFT:

$$y[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \hat{y}[n], \quad \hat{y}[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, y[k].$$

Thus, the functional

$$E_y(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left| \xi[n]\, u^{2r}[n] - \hat{y}[n] \right|^2. \tag{5.4}$$

Combining Eqs. (5.3) and (5.4), we get

$$J_\rho(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \rho\, |\xi[n]|^2\, w^{2r}[n] + \left| \xi[n]\, u^{2r}[n] - \hat{y}[n] \right|^2. \tag{5.5}$$

Proposition 5.3 *The minimum of the functional $J_\rho(S)$ on the spline space ${}^{2r}\tilde{\mathscr{S}}$, which is the minimum on $\tilde{\mathscr{F}}^r$, is provided by the spline*

$$S_\rho^{2r}[\mathbf{y}](t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n](\rho)\,\zeta^{2r}[n](t), \quad \xi[n](\rho) = \frac{\hat{y}[n]}{A_\rho^r[n]} \tag{5.6}$$
$$A_\rho^r[n] \stackrel{\text{def}}{=} \rho\, w^{2r}[n] + u^{2r}[n].$$

Proof Represent $\xi[n] = x\, e^{i\alpha}$, where $x = |\xi[n]|,\ \alpha = \arg(\xi[n])$, and $\hat{y}[n] = Y\, e^{i\beta}$, where $Y = |\hat{y}[n]|,\ \beta = \arg(\hat{y}[n])$. Then, we have

$$\rho\,|\xi[n]|^2\, w^{2r}[n] + \left|\xi[n]\, u^{2r}[n] - \hat{y}[n]\right|^2 = F(x,\alpha), \text{ where}$$
$$F(x,\alpha) = x^2 u^{2r}[n]\, A_\rho^r[n] - 2xY\, u^{2r}[n]\cos(\alpha-\beta) + Y^2.$$

From the equations

$$F'_x(x,\alpha) = u^{2r}[n]\left(2x\, A_\rho^r[n] - 2\,Y\, u^{2r}[n]\,\cos(\alpha-\beta)\right) = 0,$$
$$F'_\alpha(x,\alpha) = 2x\,Y\,u^{2r}[n]\,\sin(\alpha-\beta) = 0$$

we derive $\alpha = \beta$ and $x = Y/A_\rho^r[n]$. ■

5.1.2 Solution of the Constrained Minimization Problem

Proposition 5.4 [2] *A unique solution to the constrained minimization problem* $\min_{g\in\mathscr{F}^r} I(g)$ *subject to* $E_y(g) \le \varepsilon^2$ *is a spline* $S_{\bar\rho}^{2r}[\mathbf{y}](t) \in {}^{2r}\mathscr{S}$ *defined in Eq. (5.6), where the parameter* $\bar\rho$ *is defined from the equation*

$$e(\rho) \stackrel{\text{def}}{=} E_y(S_\rho^{2r}) = \varepsilon^2. \tag{5.7}$$

Proof The discrepancy functional for the parameterized spline $S_\rho^{2r}[\mathbf{y}](t)$ is

$$e(\rho) \stackrel{\text{def}}{=} E_y(S_\rho^{2r}) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\left|\xi[n](\rho)\,u^{2r}[n] - \hat{y}[n]\right|^{2r}$$
$$= \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\left|\hat{y}[n]\right|^2\left(\frac{\rho\, w^{2r}[n]}{\rho\, w^{2r}[n] + u^{2r}[n]}\right)^2. \tag{5.8}$$

The derivative

$$e'(\rho) \stackrel{\text{def}}{=} \frac{2}{N}\sum_{n=-N/2}^{N/2-1}\left|\hat{y}[n]\right|^2 \frac{\rho\left(w^{2r}[n]\right)^2 u^{2r}[n]}{\left(\rho\, w^{2r}[n] + u^{2r}[n]\right)^3} > 0$$

for all positive ρ. Thus, the function $e(\rho)$ grows monotonically when ρ increases and

$$e(0) = 0, \quad \lim_{\rho\to\infty} e(\rho) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \left|\hat{y}[n]\right|^2 = \sum_{k=0}^{N-1} y[k]^2. \tag{5.9}$$

Therefore, Eq. (5.7) has the unique solution $\rho = \bar{\rho}$.

Denote $i(\rho) \stackrel{\text{def}}{=} I(S_\rho^{2r}[\mathbf{y}]) = \int_0^N \left((S_\rho^{2r}[\mathbf{y}](t))^{(r)}\right)^2 dt$. Equations (5.3) and (5.6) imply that

$$i(\rho) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} |\xi_\rho[n]|^2\, w^{2r}[n]\, u^{2r}[n] = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \left|\hat{y}[n]\right|^2 \frac{u^{2r}[n\, w^{2r}[n]}{\left(\rho\, w^{2r}[n] + u^{2r}[n]\right)^2}.$$

Obviously, the function $i(\rho)$ decays monotonically, while $\lim_{\rho\to\infty} i(\rho) = 0$. Therefore, the minimum of the functional $I(S_\rho^{2r}[\mathbf{y}])$ under the condition $E_y(S_\rho^{2r}[\mathbf{y}]) \le \varepsilon^2$ is achieved when $\rho = \bar{\rho}$. Thus the spline $S_\rho^{2r}[\mathbf{y}](t)$ is the unique solution of the constraint minimization problem. ■

Comments:

1. Equation (5.9) means that, when $\rho = 0$, the spline $S_\rho^{2r}[\mathbf{y}](t)$ interpolates the data vector **y.** In the other limit case $\rho = \infty$, the spline $S_\rho^{2r}[\mathbf{y}](t)$ interpolates the zero vector. Since the interpolating spline is unique, $S_\infty^{2r}[\mathbf{y}](t) \equiv 0$. Thus, the approximation of the available data deteriorates as ρ is growing.

2. On the other hand, when $\rho = 0$,

$$i(0) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \left|\hat{y}[n]\right|^2 \frac{w^{2r}[n]}{\left(u^{2r}[n]\right)^2} = \int_0^N \left((S_0^{2r}[\mathbf{y}](t))^{(r)}\right)^2 dt$$

   is equal to the squared norm of the $r-$th derivative of the spline, which interpolates the data **y**. Apparently, when errors in the samples $y[k]$ present, this derivative is far from being regular and $i(0)$ is relatively large. When ρ is growing, the norm of the $r-$th derivative of the spline $S_\rho^{2r}[\mathbf{y}(t)$ decreases: thus the spline is becoming more regular until degradation to $S_\infty^{2r}[\mathbf{y}](t) \equiv 0$.

3. The parameter $\bar{\rho}$ found from Eq. (5.7) establishes an optimal trade-off between the approximation of noised data and the regularity of the solution. When the errors do not present, that is $\varepsilon = 0$, then the parameter is $\bar{\rho} = 0$ and the grid samples are $S_0^{2r}[\mathbf{y}](k) = y[k] = f(k)$. Thus, the smoothing spline reduces to an interpolating one. The MATLAB function `defroPSm.m` derives an optimal value of the parameter $\bar{\rho}$ by solving Eq. (5.7).

4. The grid samples of the spline $S_{\bar{\rho}}^{2r}[y]$ are

$$S_{\bar{\rho}}^{2r}[\mathbf{y}](k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi[n](\bar{\rho})\,\omega^{kn}\,u^{2r}[n] = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \omega^{kn}\,\frac{\hat{y}[n]\,u^{2r}[n]}{\bar{\rho}w^{2r}[n] + u^{2r}[n]}.$$

The samples are readily calculated by the IDFT. The spline $S_{\bar{\rho}}^{2r}(t)$ can be regarded as a spline that interpolates the signal $\mathbf{s} = \left\{S_{\bar{\rho}}^{2r}(k)\right\}$. The values between grid points are calculated by the subdivision methods that are described in Chap. 6.

5. The smoothing splines including the subdivision are computed by the MATLAB function `smoothsplaPt.m`.

5.1.3 Generators of Smoothing Splines

Denote

$$L_{\rho}^{2r}(t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{1}{A_{\rho}^{r}[n]}\,\zeta^{2r}[n](t) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1} \tau_{\rho}[n]\,g^{2r}[n](t),$$

where the sequence $\left\{\tau_{\rho}[n] \stackrel{\text{def}}{=} \sqrt{u^{4r}[n]}/A_{\rho}^{r}[n]\right\}$ is the SHA spectrum of the spline $L_{\rho}^{2r}(t) \in {}^{2r}\mathscr{S}$. Obviously, the periodic sequence $\tau_{\rho}[n]$ is strictly positive. Therefore, Proposition 4.4 claims that the spline can serve as a generator of the space ${}^{2r}\mathscr{S}$.

Proposition 5.5 *The smoothing spline $S_{\rho}^{2r}[\mathbf{y}](t)$ given in Eq. (5.6) can be represented as*

$$S_{\rho}^{2r}[\mathbf{y}](t) = \sum_{k=0}^{N-1} y[k]\,L_{\rho}^{2r}(t-k).$$

Proof Using the shift property Eq. (4.11) of the exponential splines, we transform Eq. (5.6) as

$$\begin{aligned} S_{\rho}^{2r}[\mathbf{y}](t) &= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{\hat{y}[n]}{A_{\rho}^{r}[n]}\,\zeta^{2r}[n](t) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\sum_{k=0}^{N-1} \omega^{-kn}\,y[k]\,\frac{\zeta^{2r}[n](t)}{A_{\rho}^{r}[n]} \\ &= \sum_{k=0}^{N-1} y[k]\,\frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{\zeta^{2r}[n](t-k)}{A_{\rho}^{r}[n]} = \sum_{k=0}^{N-1} y[k]\,L_{\rho}^{2r}(t-k). \end{aligned}$$

■

Note that $A_{\rho}^{r}[n]|_{\rho=0} = u^{p}[n]$. Hence, it follows that $L_{0}^{2r}(t) = L^{2r}(t)$, which is the fundamental generator of the space ${}^{2r}\mathscr{S}$. The values of the spline at grid points

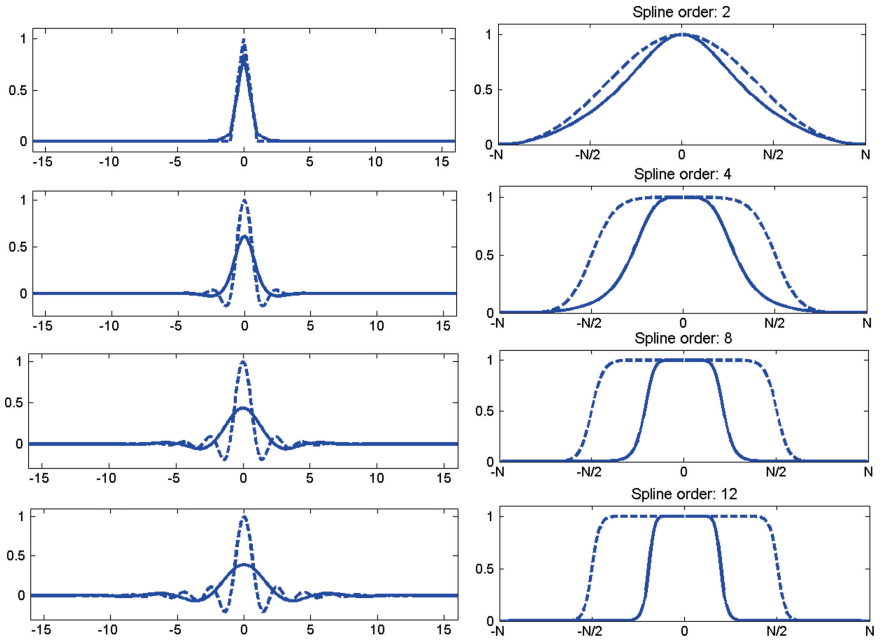


Fig. 5.1 *Left: Dashed lines* fundamental splines $L^{2r}(t)$ of orders 2, 4, 8 and 12 ($\rho = 0$). *Solid lines* smoothing generators $L_\rho^{2r}(t)$ of the same orders, $\rho = 0.1$. *Right: Dashed lines* Fourier spectra of the fundamental splines $L^{2r}(t)$. *Solid lines* Fourier spectra of the smoothing generators $L_\rho^{2r}(t)$

$$L_\rho^{2r}(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \frac{u^p[n]}{\rho\, w^{2r}[n] + u^p[n]}.$$

Thus, the grid samples of the smoothing spline $s_\rho[k] \stackrel{\text{def}}{=} S_\rho^{2r}[\mathbf{y}](k),\ m \in \mathbb{Z}$, are

$$s_\rho[k] = \sum_{l=0}^{N-1} y[l]\, L_\rho^{2r}(k-l), \quad \hat{\mathbf{s}}_\rho[n] = \frac{u^{2r}[n]}{\rho\, w^{2r}[n] + u^{2r}[n]}\, \hat{y}[n].$$

Transform of a signal $\mathbf{y} = \{y[k]\} \longrightarrow \mathbf{s}_\rho = \big\{s_\rho[k]\big\}$ can be regarded as low-pass p-filtering of the signal $\mathbf{y}$ by the parameterized p-filter $\mathbf{h}_\rho^{2r}$, whose frequency response

$$\hat{h}_\rho^{2r}[n] = \frac{u^{2r}[n]}{\rho\, w^{2r}[n] + u^{2r}[n]} = 1 - \rho \frac{w^{2r}[n]}{u^{2r}[n]} + \rho^2 \left(\frac{w^{2r}[n]}{u^{2r}[n]}\right)^2 + \dots . \tag{5.10}$$

The widths of the pass-band is controlled by the parameter ρ.

Figure 5.1, which was produced by the MATLAB code `GensplConsPF_examp.m`, displays smoothing generators of orders 2, 4, 8 and 12 with the regularization parameter $\rho = 0.1$ versus their interpolating

counterparts ($\rho = 0$) and the respective Fourier spectra. The smoothing generators are obviously smoother compared to the interpolating generators. Oscillations are significantly reduced. The low-frequency parts of the spectra remain intact while the high-frequency parts are suppressed. It stems from the structure of the denominator $A_\rho^r[n] = \rho\, w^{2r}[n] + u^p[n]$ in Eq. (5.10). The function $w^{2r}[n] \stackrel{\text{def}}{=} (2 \sin \pi n/N)^{2r} \approx 0$ when the frequency n is small, especially for splines of higher orders. Thus, when n is small, $A_\rho^r[n] \approx u^p[n]$ and the spectrum of $L_\rho^{2r}(t)$ is very close to the spectrum of $L^{2r}(t)$.

5.2 Two-Dimensional Smoothing Splines

The two-dimensional N-periodic splines from the space ${}^{p,q}\mathscr{S}$ are represented as

$$S(x, y) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} s[k, n]\, B^p(x-k)\, B^q(y-n) = \frac{1}{N} \sum_{\kappa,\nu=0}^{N-1} \xi[\kappa, \nu]\, \zeta^p[\kappa](x)\, \zeta^q[\nu](y).$$

Grid samples of the spline S are given in Eq. (4.61), and the spline $S(x, y) \in {}^{p,q}\mathscr{S}$, which interpolates an N-periodic array $\mathbf{z} \stackrel{\text{def}}{=} \{z[k, n]\}$, that is $S(k, n) = z[k, n]$, is represented as

$$S(x, y) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} e^{2\pi i (k\kappa + n\nu)} \frac{\hat{z}[\kappa, \nu]}{u^p[\kappa]\, u^q[\nu]}\, \zeta^p[\kappa](x)\, \zeta^q[\nu](y),$$

where $\hat{z}[\kappa, \nu]$ is the 2D DFT of the array $\mathbf{z} \stackrel{\text{def}}{=} \{z[k, n]\}$. The design of the two-dimensional smoothing splines is implemented by the same steps as in the 1D design. For simplicity, assume that the splines have the same order in both x and y directions. Assume that the function $f(x, y)$ to be approximated is N-periodic in both direction, and the functional

$$I(f) \stackrel{\text{def}}{=} \int_0^N \left((f_x^{(r)}(x, y))^2 + (f_y^{(r)}(x, y))^2 \right) dx\, dy < \infty.$$

Assume that the available data array is $\mathbf{z} = \left\{ z[k, n] = f(k, n) + e_{k,n} \right\},\ k, n = 0, \ldots, N-1$, and $\mathbf{e} = \left\{ e_{k,n} \right\},\ k, n = 0, \ldots, N-1$, is the array of random errors that is assumed to be a zero-mean white noise. The estimated squared norm of errors is $\varepsilon^2 = \sum_{k,n=0}^{N-1} e_{k,n}^2$.

Denote by $S_\rho(x, y) \in {}^{2r,2r}\mathscr{S}$ the spline, which minimizes the parameterized functional $J_\rho(S) \stackrel{\text{def}}{=} \rho I(S) + E_z(S)$ on the space ${}^{2r,2r}\mathscr{S}$, where the discrepancy functional $E_z(S)$ is defined as

$$E_z(S) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} (S(k,n) - z[k,n])^2.$$

Equation (4.61) implies that

$$E_z(S) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \left| \xi[\kappa,\nu]\, u^{2r}[\kappa]\, u^{2r}[\nu] - \hat{z}[\kappa,\nu] \right|^2. \tag{5.11}$$

The functional

$$\begin{aligned} I(S) &= \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \left(W^r[\kappa,\nu] + W^r[\nu,\kappa] \right) |\xi[\kappa,\nu]|^2, \\ W^r[\kappa,\nu] &\stackrel{\text{def}}{=} \left(2 \sin \frac{\pi\kappa}{N} \right)^{2r} u^{2r}[\kappa]\, u^{4r}[\nu]. \end{aligned} \tag{5.12}$$

By using Eqs. (5.11) and (5.12), the spline is explicitly represented as

$$\begin{aligned} S_\rho(x,y) &= \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \xi[\kappa,\nu](\rho)\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y), \\ \xi[\kappa,\nu](\rho) &\stackrel{\text{def}}{=} \frac{\hat{z}[\kappa,\nu]\, u^{2r}[\kappa]\, u^{2r}[\nu]}{A^r[\kappa,\nu](\rho)}, \\ A^r[\kappa,\nu](\rho) &\stackrel{\text{def}}{=} \rho \left(W^r[\kappa,\nu] + W^r[\nu,\kappa] \right) + (u^{2r}[\kappa]\, u^{2r}[\nu])^2. \end{aligned} \tag{5.13}$$

Similarly to the 1D case, an optimal regularization parameter ρ is derived from the equation $e(\rho) \stackrel{\text{def}}{=} E_z(S_\rho) = \varepsilon^2$, which is expressed as

$$e(\rho) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \left| \hat{z}[\kappa,\nu] \right|^2 \left(\frac{(\rho\ (W^r[\kappa,\nu] + W^r[\nu,\kappa])}{A^r[\kappa,\nu](\rho)} \right)^2 = \varepsilon^2, \tag{5.14}$$

and has a unique solution $\rho = \bar{\rho}$. The grid values of the spline $S_{\bar{\rho}}(x,y)$, which is called the 2D smoothing spline, are

$$S_{\bar{\rho}}(k,n) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} e^{2\pi i (k\kappa + n\nu)} \frac{\hat{z}[\kappa,\nu]\, (u^{2r}[\kappa]\, u^{2r}[\nu])^2}{\bar{\rho}\, (W^r[\kappa,\nu] + W^r[\nu,\kappa]) + (u^{2r}[\kappa]\, u^{2r}[\nu])^2}. \tag{5.15}$$

The grid values are calculated by the 2D IDFT. Certainly, when $\varepsilon^2 = 0$, the parameter $\bar{\rho} = 0$ and the spline $S_{\bar{\rho}}(x,y)$ interpolates the initial function $f(x,y)$ such that $S_0(k,n) = f(k,n)$.

The values of the splines in the internal points between the grid points are calculated by the subdivision methods that are described in Chap. 6.

The 2D smoothing splines including the subdivision are computed by the MATLAB function `smooth_spline2DPt.m`.

References

1. E.T. Whittaker, On a new method of graduation. Proc. Edinb. Math. Soc. **41**, 63–75 (1922)
2. I.J. Schoenberg, Spline functions and the problem of graduation. Proc. Nat. Acad. Sci. U.S.A. **52**(4), 947–950 (1964)
3. C.H. Reinsch, Smoothing by spline functions. Numer. Math. **10**, 177–183 (1967)
4. J.H. Ahlberg, E.N. Nilson, J.L. Walsh, *The theory of splines and their applications* (Academic Press, New York, 1987)
5. J.C. Holladay, A smoothest curve approximation. Math. Tables Aids Comput. **11**, 233–243 (1957)

Chapter 6
Calculation of Splines Values by Subdivision

Abstract Assume, the samples of a spline $S(t) \in {}^{p}\mathscr{S}$ on the grid $\mathbf{g} = \{k\}_{k\in\mathbb{Z}}$ are available: $S(k) = y[k]$. Subdivision schemes are proposed to calculate the spline's values at dyadic and triadic rational points $S(k/2^m)$ and $S(k/3^m)$. The SHA technique provides fast and explicit implementation of the subdivision for one- and two-dimensional periodic splines.

Throughout the chapter, we use some notations, which differ from the respective notations in the rest of the volume.

We denote $N = 2^j$, $N_m = 2^m N = 2^{j+m}$, $\tilde{N}_m = 3^m N$, $m \in \mathbb{N}$, and $\omega = e^{2\pi i/N}$. The space of N-periodic discrete-time signals is denoted by $\Pi[N]$. The spaces of N_m-periodic and $\tilde{N}_m$-periodic signals are denoted as $\Pi[N_m]$ and $\Pi[\tilde{N}_m]$, respectively. The direct and inverse discrete Fourier transform (DFT) of a signal $\mathbf{x} = \{x[k]\} \in \Pi[N_m]$ are

$$\hat{x}[n]_m = \sum_{k=0}^{N_m-1} \omega^{-2^{-m}kn}\, x[k], \quad x[k] = \frac{1}{N_m} \sum_{n=0}^{N_m-1} \omega^{2^{-m}kn}\, \hat{x}[n]_m. \tag{6.1}$$

If $\mathbf{x} \in \Pi[\tilde{N}_m]$ then its DFT and IDFT are

$$\hat{x}[n]_{\tilde{m}} = \sum_{k=0}^{\tilde{N}_m-1} \omega^{-3^{-m}kn}\, x[k], \quad x[k] = \frac{1}{\tilde{N}_m} \sum_{n=0}^{\tilde{N}_m-1} \omega^{3^{-m}kn}\, \hat{x}[n]_{\tilde{m}}. \tag{6.2}$$

For N-periodic signals ($m = 0$), the standard notation $\hat{x}[n] = \hat{x}[n]_0$ is retained.

The inner product and the norm in the space $\Pi[N_m]$ are defined as

$$\langle \mathbf{x}, \mathbf{y} \rangle \stackrel{\text{def}}{=} \sum_{k=0}^{N_m-1} x[k]\, y^*[k] = \frac{1}{N_m} \sum_{n=0}^{N_m-1} \hat{x}[n]_m\, \hat{y}[n]^*_m, \tag{6.3}$$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_6,

$$\|\mathbf{x}\|^2 \stackrel{\text{def}}{=} \sum_{k=0}^{N_m-1} |x[k]|^2 = \frac{1}{N_m} \sum_{n=0}^{N_m-1} |\hat{x}[n]_m|^2$$

and similarly in the space $\Pi[\tilde{N}_m]$. Here, $\cdot^*$ means complex conjugation.

6.1 Interpolatory Subdivision

Interpolatory subdivision schemes are refinement rules, which iteratively refine the data by inserting values corresponding to intermediate points, using linear combinations of values in initial points, while the data in these initial points are retained. Non-interpolatory schemes also update the initial data, in addition to the insertion of values into intermediate points. Stationary schemes use the same insertion rule at each refinement step. A scheme is called uniform if its insertion rule does not depend on the location in the data.

To be more specific, a univariate stationary uniform subdivision scheme with binary refinement [4–6] $\mathfrak{S}_a^2$ consists of the following: A function $F(t)$ that is defined on the grid $\mathbf{g}^m = \{k/2^m\}_{k\in\mathbb{Z}}$: $F(k/2^m) = f^m[k]$, is extended onto the grid $\mathbf{g}^{m+1}$ by filtering the upsampled array

$$(\uparrow 2)\mathbf{f}^m = \{\check{f}^m[k]\}, \quad \check{f}^m[k] = \begin{cases} f^m[l], & \text{if } k = 2l; \\ 0, & \text{otherwise,} \end{cases}$$

by an interpolaing filter $\mathbf{a} = \{a[k]\}_{k\in\mathbb{Z}}$ such that $a[2k] = \delta[k]$. Thus, one dyadic refinement step is

$$F(k/2^{m+1}) = f^{m+1}[k] = \sum_{l\in\mathbb{Z}} a[k-2l] f^m[l] \tag{6.4}$$
$$\Longleftrightarrow \begin{cases} f^{m+1}[2k] = f^m[k], \\ f^{m+1}[2k+1] = \sum_{l\in\mathbb{Z}} a[2(k-l)+1] f^m[l]. \end{cases}$$

The next refinement step employs $\mathbf{f}^{m+1}$ as initial data. The filter $\mathbf{a} = \{a[k]\}_{k\in\mathbb{Z}}$ is called the refinement mask of $\mathfrak{S}_a^2$.

A subdivision scheme with ternary refinement $\mathfrak{S}_a^3$ consists of the following: A function $F(t)$ that is defined on the grid $\mathbf{g}^m = \{k/3^m\}_{k\in\mathbb{Z}}$: $F(k/3^m) = f^m[k]$, is extended onto the grid $\mathbf{g}^{m+1}$ by filtering the upsampled array

$$(\uparrow 3)\mathbf{f}^m = \{\check{f}^m[k]\}, \quad \check{f}^m[k] = \begin{cases} f^m[l], & \text{if } k = 3l; \\ 0, & \text{otherwise.} \end{cases}$$

Thus, one triadic refinement step is

$$F(k/3^{m+1}) = f^{m+1}[k] = \sum_{l\in\mathbb{Z}} a[k-3l] f^m[l] \tag{6.5}$$
$$\Longleftrightarrow \begin{cases} f^{m+1}[3k] = f^m[k], \\ f^{m+1}[3k \pm 1] = \sum_{l\in\mathbb{Z}} a[3(k-l) \pm 1] f^m[l]. \end{cases}$$

The next refinement step employs $\mathbf{f}^{m+1}$ as initial data.

Non-stationary subdivision schemes use different filters for different refinement steps. In the periodic case, the periodicity of the data $\mathbf{f}^m$ depends on m. Therefore the p-filters at each step should be different, although their structure is similar.

6.2 Binary Subdivision for Periodic Splines

Assume, the samples of an N-periodic spline $S(t) \in {}^p\mathcal{S}$, which is constructed on the grid $\mathbf{g} = \{k\},\ k \in \mathbb{Z}$, are available: $S(k) = y[k]$. In this case, utilization of the SHA techniques makes computations straightforward for splines of arbitrary even orders.

6.2.1 Spline Spaces on Different Dyadic Resolution Scales

So far, we operated with the splines from the spaces ${}^p\mathcal{S}$, whose nodes are located on the grid $\{k\}$. In this Section we introduce an embedded set of periodic splines spaces corresponding to different resolution scales. Denote by ${}^p\mathcal{S}_m$ the space of $N-$periodic splines of order p on the grid $\left\{2^{-m}\, k\right\},\ k \in \mathbb{Z},\ m \le M$. For the space ${}^p\mathcal{S}_0$, the notation ${}^p\mathcal{S}$ is retained. Obviously, the space ${}^p\mathcal{S}_m$ is the subspace of ${}^p\mathcal{S}_{m+1}$. Thus, we get a set of periodic splines spaces that correspond to different resolution scales such that ${}^p\mathcal{S} \subset {}^p\mathcal{S}_1 \ldots \subset {}^p\mathcal{S}_m \ldots$.

The N-periodic normalized B-spline of first order on the grid $\left\{2^{-m}\, k\right\}$ and its Fourier coefficients are

$$B_m^1(t) \stackrel{\text{def}}{=} \sum_{k\in\mathbb{Z}} 2^m \chi[-2^{-m-1}, 2^{-m-1}]\,(t + kN)\,, \quad c_n(B_m^1) = \frac{\sin \pi n/N_m}{\pi n/N_m},$$

where the characteristic function $\chi[a,b](t)$ of the interval $[a,b]$ is defined in Eq. (3.12) and $N_m = N\,2^m$.

The N-periodic normalized B-spline of order p is defined via the iterated circular convolution

$$B_m^p(t) \stackrel{\text{def}}{=} B_m^1 \circledast B_m^{p-1}(t) = \frac{1}{N} \sum_{n\in\mathbb{Z}} \left(\frac{\sin(\pi n/N_m)}{\pi n/N_m} \right)^p e^{2\pi i n t/N}. \tag{6.6}$$

Each spline $S_m(t) \in {}^p\mathscr{S}_m$ is represented by

$$S_m(t) = \sum_{k=0}^{N_m-1} q[k]\, B_m^p\left(t - 2^{-m}k\right) = \frac{1}{N_m} \sum_{n=-N_m/2}^{N_m/2-1} \xi[n]\, \zeta_m^p[n](t), \quad \xi[n] = \hat{q}[n]_m, \tag{6.7}$$

where the exponential splines, which form orthogonal bases of ${}^p\mathscr{S}_m$, are the Zak transforms of the B-splines:

$$\begin{aligned} \zeta_m^p[n](t) &\stackrel{\text{def}}{=} \sum_{k=0}^{N_m-1} \omega^{2^{-m}nk} B_m^p(t - 2^{-m}k) \\ &= 2^m \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+2^{-m}l)t} \left(\frac{\sin \pi(n/N_m + l)}{\pi(n/N_m + l)}\right)^p, \end{aligned} \tag{6.8}$$

where $n = -N_m/2, \ldots, N_m/2-1$. The sequence $\left\{\zeta_m^p[n](t)\right\}$ of mutually orthogonal splines is N_m-periodic: $\zeta_m^p[n](t) = \zeta_m^p[n + N_m](t)$. The N_m-periodic characteristic sequence of the space ${}^p\mathscr{S}_m$ is

$$u_m^p[n] \stackrel{\text{def}}{=} \zeta_m^p[n](0) = \sum_{k=0}^{N_m-1} \omega^{-2^{-m}nk} B_m^p(2^{-m}k) = 2^m \sum_{l\in\mathbb{Z}} \left(\frac{\sin \pi(n/N_m + l)}{\pi(n/N_m + l)}\right)^p. \tag{6.9}$$

The spline $S_m(t) \in {}^p\mathscr{S}_m$, which interpolaes an N_m-periodic signal $\mathbf{x} = \{x[k]\}$ by $S(2^{-m}k) = x[k],\ k \in \mathbb{Z}$, is

$$S_m(t) = \frac{1}{N_m} \sum_{n=0}^{N_m-1} \frac{\hat{x}[n]_m}{u_m^p[n]}\, \zeta_m^p[n](t), \quad \hat{x}[n]_m = \sum_{k=0}^{N_m-1} \omega^{-2^{-m}nk} x[k]. \tag{6.10}$$

Remark 6.2.1 *The previous notations $\zeta^p[n](t) \equiv \zeta_0^p[n](t)$, $u^p[n] \equiv u_0^p[n]$ are retained for splines on the initial scale ($m = 0$).*

Proposition 6.1 *The characteristic sequence $u_m^p[n]$ of the space ${}^p\mathscr{S}_m$ can be calculated via the $2^m N-$point DFT of the sampled B-spline $B^p(t) \in {}^p\mathscr{S}$.*

Proof At the initial scale the identity holds

$$\sum_{k=0}^{N-1} \omega^{-nk} B^p(k) = \sum_{l\in\mathbb{Z}} \left(\frac{\sin \pi(n/N + l)}{\pi(n/N + l)}\right)^p. \tag{6.11}$$

Replace in Eq. (6.11) N by $N_m = 2^m N$ and compare the result with Eq. (6.9). Then,

$$\sum_{k=0}^{N_m-1} e^{-2\pi i n k/N_m} B^p(k) = \sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi\,(n/N_m+l)}{\pi\,(n/N_m+l)}\right)^p = 2^{-m}\,u_m^p[n].$$

Thus, the sequence $u_m^p[n]$ is the $2^m N-$point DFT of the sampled B-spline $\left\{ b^p[k] \stackrel{\text{def}}{=} B^p(k) \right\}$ multiplied by 2^m, to become

$$u_m^p[n] = 2^m\,\hat{b}^p[n]_m = 2^m \sum_{k=0}^{N_m-1} \omega^{-2^{-m}kn}\,B^p(k). \tag{6.12}$$

■

6.2.2 Insertion Rule

Calculation of values of even order splines at dyadic rational points is implemented via the following refinement steps:

Dyadic periodic spline insertion rule

Assume, a spline $S_0(t)$ belongs to the space ${}^p\mathcal{S}$ and the samples $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{ f^0[k] = S_0(k) \right\}$, $k \in \mathbb{Z}$ are available. For $m = 1, 2, \ldots$, construct a spline $S_{m-1}(t) \in {}^p\mathcal{S}_{m-1}$, which interpolaes the sequence $\mathbf{f}^{m-1} \stackrel{\text{def}}{=} \left\{ f^{m-1}[k] \right\}$ on the grid $\mathbf{g}^{m-1} \stackrel{\text{def}}{=} \left\{ k/2^{m-1} \right\}$, $k \in \mathbb{Z}$: $S_{m-1}\left(k/2^{m-1}\right) = f^{m-1}[k]$, $k \in \mathbb{Z}$. Then, $f^m[k] \stackrel{\text{def}}{=} S_{m-1}\left(k/2^m\right)$, $k \in \mathbb{Z}$.

In other words, in order to refine the data $\mathbf{f}^{m-1}$ from the grid $\left\{ k/2^{m-1} \right\}$ to the grid $\{k/2^m\}$, we construct the spline $S_{m-1}(t)$, which interpolaes $\mathbf{f}^{m-1}$ on the grid $\left\{ k/2^{m-1} \right\}$ and define $f^m[2k] \stackrel{\text{def}}{=} f^{m-1}[k]$, $\quad f^m[2k+1] \stackrel{\text{def}}{=} S_{m-1}((k+1/2)/2^{m-1})$, which are the spline values at midpoints between the interpolaion points.

If the order of the splines $S_m(t) \in {}^{2r}\mathcal{S}_m$ is even, then this insertion rule reproduces the spline, which means that $S_m(t) \equiv S_{m-1}(t) \equiv \cdots \equiv S_0(t)$. Consequently, each refinement step provides values of the spline $S_0(t)$ at the subsequent set of dyadic rational points.

This is not the case with the splines of odd order. The subdivision scheme with the dyadic spline insertion rule with an odd-order spline converges to a function, which is smoother than the initial spline [6].

Theorem 6.1 *Let $S(t) \in {}^{2r}\mathscr{S}$ be an N-periodic spline of order $2r$ with nodes on the grid $\{k\},\ k \in \mathbb{Z}$, and its samples $\left\{S(k) = f^0[k]\right\}$, $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{f^0[k]\right\},\ k \in \mathbb{Z}$. Then, all the subsequent subdivision steps with the dyadic spline insertion rule reproduce the values $\{f^m[k] = S(k/2^m)\},\ k \in \mathbb{Z},\ m = 1, 2, \ldots$ of this spline.*

Proof Due to the minimal norm property (Proposition 5.1), we get

$$\mu \stackrel{\text{def}}{=} \int_0^N |S^{(r)}(t)|^2\, dt \le \int_0^N |g^{(r)}(t)|^2\, dt,$$

where $g(t)$ is any $N-$periodic function such that the derivative $g^{(r)}(t)$ is square integrable on the interval $[0, N]$ and $\left\{g(k) = f^0[k]\right\},\ k \in \mathbb{Z}$. Let $S_1(t) \in {}^{2r}\mathscr{S}_1$ be a spline of order $2r$, which interpolaes the values $\{f^1[k] = S(k/2)\}_{k\in\mathbb{Z}}$. Then,

$$\nu \stackrel{\text{def}}{=} \int_0^N |S_1^{(r)}(t)|^2\, dt \le \int_0^N |G^{(r)}(t)|^2\, dt,$$

where $G(t)$ is any function such that $G^{(r)}(t)$ is square integrable and $G(k/2) = f^1[k] = S(k/2)$. Hence, $\nu \le \mu$. On the other hand, $S_1(k) = f^0[k]$, therefore, $\mu \le \nu$. Thus $\int_0^N |S^{(r)}(t)|^2\, dt = \int_0^N |S_1^{(r)}(t)|^2\, dt$. Hence, it follows that $S_1(t) \equiv S(t)$. Repeating the above reasoning, we find that the spline $S_2(t) \in {}^{2r}\mathscr{S}_2$, which interpolaes the values $\{f^2[k] = S_1(k/4)\}_{k\in\mathbb{Z}}$, coincides with $S_1(t)$. Consequently, it coincides with $S(t)$. The same is true for any spline $S_m(t) \in {}^{2r}\mathscr{S}_{2^m}$, which interpolaes the values $\{f^m[k] = S_{m-1}(k/2^m)\}_{k\in\mathbb{Z}}$. ■

6.2.3 Periodic Spline Filters for Binary Subdivision

The spline $S_0(t) \in {}^{p}\mathscr{S}$, which interpolaes an N-periodic sequence $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{f^0[k]\right\}$ such that $S(k) = f^0[k],\ k \in \mathbb{Z}$, is represented by orthogonal exponential splines

$$\begin{aligned} S_0(t) &= \frac{1}{N}\sum_{n=0}^{N-1} \frac{\hat{f}^0[n]}{u^p[n]}\, \zeta^p[n](t), \quad \hat{f}^0[n] = \sum_{k\in\mathbb{Z}} \omega^{-kn} f^0[k], \\ \zeta^p[n](t) &= \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+l)t} \left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p, \\ u^p[n] &= \zeta^p[n](0) = \sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p. \end{aligned} \tag{6.13}$$

The DFT of the refined array can be split into the polyphase components

$$\hat{f}^1[n]_1 = \sum_{k=0}^{2N-1} \omega^{-kn/2}\, f^1[k] = \hat{f}_0^1[n] + \omega^{-n/2}\, \hat{f}_1^1[n], \tag{6.14}$$

$$\hat{f}_0^1[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn}\, f^1[2k] = \hat{f}^0[n], \quad \hat{f}_1^1[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn}\, f^1[2k+1].$$

According to the dyadic insertion rule and to the shift property Eq. (4.1) of the exponential splines, we get

$$\begin{aligned} f^1[2k+1] &= S_0\left(k+\frac{1}{2}\right) = \frac{1}{N}\sum_{n=0}^{N-1} \frac{\hat{f}^0[n]}{u^p[n]}\, \zeta^p[n]\left(k+\frac{1}{2}\right), \\ \zeta^p[n]\left(k+\frac{1}{2}\right) &= \omega^{kn}\, \omega^{n/2}\, \upsilon^p[n], \\ \upsilon^p[n] &\stackrel{\text{def}}{=} \omega^{-n/2}\, \zeta^p[n]\left(\frac{1}{2}\right) = \sum_{l\in\mathbb{Z}} e^{\pi i l} \left(\frac{\sin \pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p. \end{aligned} \tag{6.15}$$

Consequently,

$$f^1[2k+1] = \frac{1}{N}\sum_{n=0}^{N-1} \frac{\hat{f}^0[n]}{u^p[n]}\, \omega^{kn}\, \omega^{n/2}\, \upsilon^p[n] \Longrightarrow \hat{f}_1^1[n] = \omega^{n/2}\frac{\upsilon^p[n]}{u^p[n]}\, \hat{f}^0[n]. \tag{6.16}$$

By substituting Eq. (6.16) into (6.14), we get

$$\hat{f}^1[n]_1 = \hat{a}_0^p[n]\, \hat{f}^0[n], \text{ where } \hat{a}_0^p[n] \stackrel{\text{def}}{=} \frac{u^p[n] + \upsilon^p[n]}{u^p[n]} \tag{6.17}$$

$$f^1[k] = \frac{1}{2N}\sum_{k=0}^{2N-1} \omega^{kn/2}\, \hat{a}_0^p[n]\, \hat{f}^0[n]. \tag{6.18}$$

Thus, the fine array $\mathbf{f}^1$ is derived by p-filtering the initial data array $\mathbf{f}^0$ by the p-filter $\mathbf{a}_0^p$, whose frequency response $\hat{a}_0^p[n]$ is defined in Eq. (6.17).

Proposition 6.2 *If the order $p = 2r$ is even, then the frequency response of the p-filter $\mathbf{a}_0^p$ is*

$$\hat{a}_0^{2r}[n] = 2\,\cos^{2r}\frac{\pi n}{2N}\,\frac{u_1^{2r}[n]}{u^{2r}[n]}, \quad n = 0, \ldots, 2N-1, \tag{6.19}$$

where $u_m^{2r}[n]$ are defined in Eq. (3.12) .

Proof Denote, $A[n] \stackrel{\text{def}}{=} u^{2r}[n] + \upsilon^{2r}[n]$, then $\hat{a}_0^{2r}[n] = A[n]/u^{2r}[n]$. By combining Eqs. (6.13) and (6.15), we get

$$A[n] = \sum_{l\in\mathbb{Z}} \left(1 + e^{i\pi l}\right) \left(\frac{\sin \pi\,(n/N + l)}{\pi\,(n/N + l)}\right)^{2r}$$
$$= \sum_{l\in\mathbb{Z}} \left(1 + (-1^l)\right) \left(\frac{\sin \pi\,(n/N + l)}{\pi\,(n/N + l)}\right)^{2r}.$$

Thus, only even terms in the series are present:

$$A[n] = 2\sum_{l\in\mathbb{Z}} \left(\frac{\sin 2\pi\,(n/2N + l)}{2\pi\,(n/2N + l)}\right)^{2r} = 2\sum_{l\in\mathbb{Z}} \left(\frac{\sin \pi\,(n/2N + l)\,\cos \pi\,(n/2N + l)}{\pi\,(n/2N + l)}\right)^{2r}$$
$$= 2\cos^{2r}\frac{\pi n}{2N} \sum_{l\in\mathbb{Z}} \left(\frac{\sin \pi\,(n/2N + l)}{\pi\,(n/2N + l)}\right)^{2r} = 2\cos^{2r}\frac{\pi n}{2N}\, u_1^{2r}[n].$$

■

Similarly to Eq. (6.17), the refinement expression from the resolution scale $m-1$ is obtained:

$$\hat{f}^m[n]_m = \hat{a}_{m-1}^{2r}[n]\,\hat{f}^{m-1}[n]_{m-1}, \quad \hat{a}_{m-1}^{2r}[n] = 2\,\cos^{2r}\frac{\pi n}{N_m}\,\frac{u_m^{2r}[n]}{u_{m-1}^{2r}[n]}, \tag{6.20}$$

where $u_m^{2r}[n]$ are defined in Eq. (6.9) and $n = 0, \ldots, N_m - 1$.

6.2.4 Computation of Periodic Splines at Dyadic Rational Points

It follows from Theorem 6.1 and Eq. (6.20) that, once the samples $\left\{S(k) = f^0[k]\right\}$, $k = 0, \ldots, N-1$ of a spline $S(t) \in {}^{2r}\mathscr{S}$ are available, its values $\{S(k/2^m)\}$, $k \in \mathbb{Z}$, are derived by m successive refinement steps with the initial data $\mathbf{f}^0$ and p-filters $\mathbf{a}_0^{2r}$, $\mathbf{a}_1^{2r}$, $\ldots$, $\mathbf{a}_{m-1}^{2r}$, whose frequency responses are given in Eq. (6.20). However, actually, the spline subdivision from the sparse grid $\{k\}$ to any fine grid $\{k/2^m\}$ can be implemented by one direct DFT, a few multiplications and one inverse DFT. This process can be described explicitly. Equation (6.20) implies that

$$\hat{f}^1[n]_1 = \hat{a}_0^{2r}[n]\,\hat{f}^0[n], \quad \hat{a}_0^{2r}[n] = 2\cos^{2r}\frac{\pi n}{2N}\,\frac{u_1^{2r}[n]}{u^{2r}[n]}\,\hat{f}^0[n],$$
$$\hat{f}^2[n]_2 = 2\,\cos^{2r}\frac{\pi n}{4N}\,\frac{u_2^{2r}[n]}{u_1^{2r}[n]}\,\hat{f}^1[n]_1 = 2^2\cos^{2r}\frac{\pi n}{4N}\,\cos^{2r}\frac{\pi n}{2N}\,\frac{u_2^{2r}[n]\,u_1^{2r}[n]}{u_1^{2r}[n]\,u^{2r}[n]}\,\hat{f}^0[n],$$
$$\hat{f}^m[n]_m = 2^m\,\frac{u_m^{2r}[n]}{u^{2r}[n]}\,\hat{f}^0[n]\,\prod_{l=1}^{m}\cos^{2r}\frac{\pi n}{2^l N}. \tag{6.21}$$

Then, computation of the values $S(k/2^m) = f^m[k]$ of the spline $S(t) \in {}^{2r}\mathscr{S}$ from the samples $S(k) = f^0[k]$ is implemented by the following steps:

1. Calculate the $N-$point DFT $\hat{f}^0[n] = \sum_{k=0}^{N-1} \omega^{-kn} f^0[k]$ of the initial data $\mathbf{f}^0 = \left\{f^0[k]\right\}$.
2. Multiply the array $\left\{\hat{f}^0[n]\right\}$

$$\hat{f}^m[n]_m = 2^m \frac{u_m^{2r}[n]}{u^{2r}[n]} \hat{f}^0[n] \prod_{l=1}^{m} \cos^{2r} \frac{\pi n}{2^l N}$$

   bearing in mind that $u^{2r}[n]$ and $\hat{f}^0[n]$ are $N-$periodic sequences, while $u_m^{2r}[n]$ is $2^m N-$periodic.
3. Implement the $2^m N-$point IDFT

$$S(k/2^m) = f^m[k] = \frac{1}{N_m} \sum_{n=0}^{N_m-1} \omega^{2^{-m}kn} \hat{f}^m[n]_m \tag{6.22}$$

   of the sequence $\hat{\mathbf{f}}^m = \left\{\hat{f}^m[n]_m\right\}$.

Calculation of the spline values at dyadic rational points is implemented by the MATLAB function `intersplliPt.m`.

6.3 Ternary Periodic Spline Subdivision

It was established in Sect. 6.2 that values of even-order splines at the dyadic rational points can be readily derived by Eq. (6.22) from the initial grid samples. This is not the case for splines of odd order. The above binary subdivision scheme being applied to an odd-order spline does not restore this spline but rather converges to a function that is smoother than the original spline [6]. However, the ternary subdivision scheme to be described restores the values of splines of any order at the triadic rational points $\left\{3^{-m}k\right\}$, $m \in \mathbb{N}$.

6.3.1 Super-Resolution Spline Spaces (Triadic Scale)

Recall that $\tilde{N}_m = 3^m N$, $m = 1, 2, \ldots$, and $\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$. The "triadic" DFT of an $\tilde{N}_m-$periodic signal $\mathbf{x} = \{x[k]\}$ is denoted as

$$\hat{x}[n]_{\tilde{m}} \stackrel{\text{def}}{=} \sum_{k=0}^{\tilde{N}_m-1} e^{-2\pi ink/\tilde{N}_m} x[k] = \sum_{k=0}^{\tilde{N}_m-1} \omega^{-3^{-m}nk} x[k], \quad x[k] = \frac{1}{\tilde{N}_m} \sum_{n=0}^{\tilde{N}_m-1} \omega^{3^{-m}nk} \hat{x}[n]_{\tilde{m}}.$$

We refine the spline space ${}^{p}\mathscr{S}$ along the triadic scale ${}^{p}\mathscr{S} = {}^{p}\tilde{\mathscr{S}}_0 \subset {}^{p}\tilde{\mathscr{S}}_1 \subset {}^{p}\tilde{\mathscr{S}}_2 \ldots \subset {}^{p}\tilde{\mathscr{S}}_m \ldots$, where ${}^{p}\tilde{\mathscr{S}}_m$ denotes the space of $N-$periodic splines of order p defined on the grid $\left\{3^{-m}k\right\}$, $k \in \mathbb{Z}$, $m = 0, 1, \ldots$.

The N-periodic normalized B-spline of the first order on the grid $\left\{3^{-m}k\right\}$ and its Fourier coefficients are

$$\tilde{B}^1_m(t) \stackrel{\text{def}}{=} \sum_{k\in\mathbb{Z}} 3^m \chi[-3^{-m-1}, 3^{-m-1}]\,(t + kN)\,, \quad c_n(\tilde{B}^1_m) = \frac{\sin \pi n/\tilde{N}_m}{\pi n/\tilde{N}_m},$$

where the characteristic function $\chi[a, b](t)$ is defined in Eq. (3.12).

The N-periodic triadic normalized B-spline of order p is defined via the iterated circular convolution

$$\tilde{B}^p_m(t) \stackrel{\text{def}}{=} \tilde{B}^1_m \circledast \tilde{B}^{p-1}_m(t) = \frac{1}{N}\sum_{n\in\mathbb{Z}} \left(\frac{\sin(\pi n/\tilde{N}_m)}{\pi n/\tilde{N}_m}\right)^p e^{2\pi i n t/N}. \tag{6.23}$$

Each spline $S(t) \in {}^{p}\tilde{\mathscr{S}}_m$ is represented as

$$S(t) = \sum_{k=0}^{\tilde{N}_m-1} q[k]\tilde{B}^p_m\left(t - 3^{-m}k\right) = \frac{1}{\tilde{N}_m}\sum_{n=0}^{\tilde{N}_m-1} \xi[n]\,\tilde{\zeta}^p_m[n](t), \quad \xi[n] = \sum_{k=0}^{\tilde{N}_m-1} e^{-2\pi i n k/\tilde{N}_m} q[k],$$

where the exponential splines, which form orthogonal bases of ${}^{p}\tilde{\mathscr{S}}_m$, are

$$\begin{aligned}\tilde{\zeta}^p_m[n](t) &\stackrel{\text{def}}{=} \sum_{k=0}^{\tilde{N}_m-1} e^{2\pi i n k/\tilde{N}_m}\,\tilde{B}^p_m(t - 3^{-m}k) \\ &= 3^m \sum_{l\in\mathbb{Z}} e^{2\pi i (n/N + 3^m l)t}\left(\frac{\sin \pi(n/\tilde{N}_m + l)}{\pi(n/\tilde{N}_m + l)}\right)^p,\end{aligned} \tag{6.24}$$

$n = 0, \ldots, \tilde{N}_m - 1$. The $\tilde{N}_m$ -periodic characteristic sequence of the space ${}^{p}\tilde{\mathscr{S}}_m$ is

$$\begin{aligned}\tilde{u}^p_m[n] &\stackrel{\text{def}}{=} \tilde{\zeta}^p_m[n](0) = \sum_{k=0}^{\tilde{N}_m-1} e^{2\pi i n k/\tilde{N}_m}\,\tilde{B}^p_m(3^{-m}k) \\ &= 3^m \sum_{l\in\mathbb{Z}}\left(\frac{\sin \pi(n/\tilde{N}_m + l)}{\pi(n/\tilde{N}_m + l)}\right)^p.\end{aligned} \tag{6.25}$$

The proof of the following fact is similar to the proof of Proposition 6.1.

Proposition 6.3 *The characteristic sequence $\tilde{u}^p_m[n]$ of the space ${}^{p}\tilde{\mathscr{S}}_m$ can be calculated via the $3^m N-$point DFT of the sampled B-spline $B^p(t) \in {}^{p}\mathscr{S}$:*

$$\tilde{u}_m^p[n] = 3^m\,\hat{b}^p[n]_{\tilde{m}} = 3^m \sum_{k=0}^{\tilde{N}_m-1} \omega^{-3^{-m}kn}\, B^p(k). \tag{6.26}$$

The spline $S(t) \in {}^p\tilde{\mathscr{S}}_m$, which interpolaes an $\tilde{N}_m$ -periodic signal $\mathbf{x} = \{x[k]\}$: $S(k/3^m) = x[k],\ k \in \mathbb{Z}$, is

$$S(t) = \frac{1}{\tilde{N}_m} \sum_{n=0}^{\tilde{N}_m-1} \frac{\hat{x}[n]_{\tilde{m}}}{\tilde{u}_m^p[n]}\, \tilde{\zeta}_m^p[n](t), \quad \hat{x}[n]_{\tilde{m}} \stackrel{\text{def}}{=} \sum_{k=0}^{\tilde{N}_m-1} \omega^{-3^{-m}nk} x[k]. \tag{6.27}$$

6.3.2 Insertion Rule

Calculation of the values at triadic rational points of N-periodic splines of order p is implemented by the following refinement steps.

Triadic periodic spline insertion rule:

Assume we have a spline $S_0(t) \in {}^p\mathscr{S}$ *and* $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{f^0[k] = S_0(k)\right\},\ k \in \mathbb{Z}$. *For* $m = 1, 2, \ldots$, *construct on the grid* $\tilde{\mathbf{g}}^{m-1} \stackrel{\text{def}}{=} \left\{k/3^{m-1}\right\},\ k \in \mathbb{Z}$, *a spline* $S_{m-1}(t) \in {}^p\tilde{\mathscr{S}}_{m-1}$, $S_{m-1}\left(k/3^{m-1}\right) = f^{m-1}[k],\ k \in \mathbb{Z}$, *which interpolaes the sequence* $\mathbf{f}^{m-1} \stackrel{\text{def}}{=} \left\{f^{m-1}[k]\right\}$. *Then,* $f^m[k] \stackrel{\text{def}}{=} S_{m-1}\left(k/3^m\right),\ k \in \mathbb{Z}$.

In other words, in order to refine the data $\mathbf{f}^{m-1}$ from the grid $\left\{k/3^{m-1}\right\}$ to the grid $\{k/3^m\}$, we construct the spline $S_{m-1}(t)$, which interpolaes $\mathbf{f}^{m-1}$ on the grid $\left\{k/3^{m-1}\right\}$ and define $f^m[3k] = f^{m-1}[k]$ and $f^m[3k \pm 1] = S_{m-1}((k \pm 1/3)/3^{m-1})$ that are values of the spline at the points around the interpolaion points.

This insertion rule reproduces a spline of any order, which means that $S_m(t) \equiv S_{m-1}(t) \equiv \ldots \equiv S_0(t)$. Consequently, each refinement step provides values of the spline $S_0(t)$ at the subsequent set of triadic rational points.

The proof of this fact for splines of even order is similar to the proof of Theorem 6.1.

6.3.3 Periodic Spline Filters for Ternary Subdivision

The spline $S_0(t) \in {}^p\tilde{\mathscr{S}}$ which interpolaes the N-periodic sequence $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{f^0[k]\right\}$ (that is $S(k) = f^0[k],\ k \in \mathbb{Z},$) is represented by

$$S_0(t) = \frac{1}{N}\sum_{n=0}^{N-1} \frac{\hat{f}^0[n]}{u^p[n]}\,\zeta^p[n](t), \quad \zeta^p[n](t) = \sum_{l\in\mathbb{Z}} e^{2\pi i(n/N+l)t}\left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p.$$

The DFT of the refined array

$$\hat{f}^1[n]_{\tilde{1}} = \hat{f}_0^1[n] + e^{-2\pi i n/3N}\,\hat{f}_1^1[n] + e^{2\pi i n/3N}\,\hat{f}_{-1}^1[n], \tag{6.28}$$
$$\hat{f}_0^1[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn}\, f^1[3k] = \hat{f}^0[n], \quad \hat{f}_{\pm1}^1[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn}\, f^1[3k\pm1].$$

According to the triadic insertion rule and to the shift property Eq. (4.11) of the exponential splines, we get

$$f^1[3k\pm1] = S_0\left(k\pm\frac{1}{3}\right) = \frac{1}{N}\sum_{n=0}^{N-1}\frac{\hat{f}^0[n]}{u^p[n]}\,\zeta^p[n]\left(k\pm\frac{1}{3}\right) \tag{6.29}$$
$$\zeta^p[n]\left(k\pm\frac{1}{3}\right) = \omega^{kn}\,e^{\pm2\pi i n/3N}\,\upsilon_{\pm1}^p[n], \quad \text{where}$$

$$\upsilon_{\pm1}^p[n] \stackrel{\text{def}}{=} e^{\mp2\pi i n/3N}\,\zeta^p[n]\left(\frac{1}{3}\right) = \sum_{l\in\mathbb{Z}} e^{\pm2\pi i l/3}\left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p. \tag{6.30}$$

Consequently,

$$f^1[3k\pm1] = \frac{1}{N}\sum_{n=0}^{N-1}\frac{\hat{f}^0[n]}{u^p[n]}\,\omega^{kn}\,e^{\pm2\pi i n/3N}\,\upsilon_{\pm1}^p[n] \tag{6.31}$$
$$\Longrightarrow \hat{\mathbf{f}}_{\pm1}^1[n] = e^{\pm2\pi i n/3N}\,\frac{\upsilon_{\pm1}^p[n]}{u^p[n]}\,\hat{f}^0[n].$$

Substituting Eq. (6.31) into (6.28), we get

$$\hat{f}^1[n]_{\tilde{1}} = \hat{a}_0^p[n]\,\hat{f}^0[n], \text{ where } \hat{a}_0^p[n] \stackrel{\text{def}}{=} \frac{u^p[n]+\upsilon_1^p[n]+\upsilon_{-1}^p[n]}{u^p[n]}, \tag{6.32}$$
$$f^1[k] = \frac{1}{3N}\sum_{k=0}^{3N-1}\omega^{kn/3}\,\hat{a}_0^p[n]\,\hat{f}^0[n].$$

Equation (6.32) means that the refined array $\mathbf{f}^1$ is derived by p-filtering of the initial array $\mathbf{f}^0$ with the p-filter $\mathbf{a}_0^p$, whose frequency response is $\{\hat{a}_0^p[n]\}$.

Proposition 6.4 *The frequency response of the p-filter* $\mathbf{a}_0^p$ *for splines of order* p *is*

$$\hat{a}_0^p[n] = 3^{-p}\left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \frac{\tilde{u}_1^p[n]}{u^p[n]}, \quad n = 0, \ldots, 3N-1, \tag{6.33}$$

where the characteristic sequences $\tilde{u}_m^p[n]$ *are defined in* Eq. (6.25)*.*

Proof Denote $A[n] \stackrel{\text{def}}{=} u^p[n] + \upsilon_1^p[n] + \upsilon_{-1}^p[n]$. Then, $\hat{a}_0^p[n] = A[n]/u^p[n]$. Combining Eqs. (6.13) and (6.30), we get

$$\begin{aligned} A[n] &= \sum_{l\in\mathbb{Z}} \left(1 + e^{2\pi i l/3} + e^{-2\pi i l/3}\right)\left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p \\ &= \sum_{l\in\mathbb{Z}} \left(1 + 2\cos\frac{2\pi l}{3}\right)\left(\frac{\sin\pi\,(n/N+l)}{\pi\,(n/N+l)}\right)^p. \end{aligned}$$

Thus, only the terms $l = 3\nu,\ \nu \in \mathbb{Z}$, in the series are present:

$$A[n] = 3\sum_{l\in\mathbb{Z}} \left(\frac{\sin 3\pi\,(n/3N+l)}{3\pi\,(n/3N+l)}\right)^p.$$

Using the trigonometric identity $\sin 3\alpha = \sin\alpha(1 + 2\cos 2\alpha)$, we get

$$\begin{aligned} A[n] &= 3^{1-p}\sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi\,(n/3N+l)\,(1+2\cos\pi\,(2n/3N+2l)}{\pi\,(n/2N+l)}\right)^p \\ &= 3^{1-p}\left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \sum_{l\in\mathbb{Z}} \left(\frac{\sin\pi\,(n/3N+l)}{\pi\,(n/3N+l)}\right)^p = \frac{1}{3^p}\left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \tilde{u}_1^p[n]. \end{aligned}$$

■

Corollary 6.1 *The DFT of the m-th refined array of splines of order* p *is*

$$\begin{aligned} \hat{f}^m[n]_{\tilde{m}} &= \hat{a}_{m-1}^p[n]\,\hat{f}^{m-1}[n]_{\widetilde{m-1}}, \\ \hat{a}_{m-1}^p[n] &= \frac{1}{3^p}\left(1 + 2\cos\frac{2\pi n}{\tilde{N}_m}\right)^p \frac{\tilde{u}_m^p[n]}{\tilde{u}_{m-1}^p[n]}, \end{aligned} \tag{6.34}$$

where $\tilde{u}_m^p[n]$ *are defined in* Eq.(6.25)*,* $\tilde{N}_m = N\,3^{-m}, \quad n = 0, \ldots, \tilde{N}_m - 1.$

6.3.4 Computation of Periodic Splines at Triadic Rational Points

Now we are in a position to justify the claim about restoration of splines by ternary subdivision.

Theorem 6.2 *Let $S_0(t) \in {}^p\mathcal{S}$ be an N-periodic spline of order p defined on the grid $\{k\}$, $k \in \mathbb{Z}$. Its samples are $\left\{S(k) = f^0[k]\right\}$, $k \in \mathbb{Z}$, where the sequence $\mathbf{f}^0 \stackrel{\text{def}}{=} \left\{f^0[k]\right\}$, $k \in \mathbb{Z}$ is N-periodic. Then, all the subsequent subdivision steps with the triadic periodic spline insertion rule reproduce the values of this spline*

$$f^m[k] = S(k/3^m), \quad k \in \mathbb{Z},\ m = 1, 2, \ldots, . \tag{6.35}$$

Proof From the definition of the triadic insertion rule we have $S_0(k/3) = f^1[k]$. The next subdivision step consists of construction of the spline $S_1(t) \in {}^p\tilde{\mathcal{S}}_1$ on the grid $\mathbf{g}^1 = \{k/3\}$ such that $S_1(k/3) = f^1[k]$. Then, by definition, $f^2[k] = S_1(k/9)$. We prove that $f^2[k] = S_0(k/9)$. The subsequent relations in Eq. (6.35) are derived by a simple induction.

The array $\mathbf{f}^2$ is obtained by successive application of the filters $\mathbf{a}_1^p$ after $\mathbf{a}_0^p$ to the array $\mathbf{f}^0$:

$$\begin{aligned} &\hat{f}^2[n]_{\tilde{2}} = \hat{a}_1^p[n]\,\hat{f}^1[n]_{\tilde{1}} = \hat{a}_1^p[n]\,\hat{a}_0^p[n]\,\hat{f}^0[n] \qquad (6.36)\\ &= \frac{1}{3^{2p}}\left(1 + 2\cos\frac{2\pi n}{9N}\right)^p \frac{\tilde{u}_2^p[n]}{\tilde{u}_1^p[n]} \left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \frac{\tilde{u}_1^p[n]}{u^p[n]}\ \hat{f}^0[n] = \hat{H}[n]\,\frac{\hat{f}^0[n]}{u^p[n]}. \end{aligned}$$

where

$$\hat{H}[n] \stackrel{\text{def}}{=} \frac{1}{3^{2p}}\left(1 + 2\cos\frac{2\pi n}{9N}\right)^p \left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \tilde{u}_2^p[n]. \tag{6.37}$$

Denote $\mathbf{s} \stackrel{\text{def}}{=} \{s[k] = S_0(k/9)\}$, $k \in \mathbb{Z}$. Similarly to Eq.(6.28), the $9N$-point DFT of this array is represented by

$$\begin{aligned} \hat{s}[n]_{\tilde{2}} &= \sum_{k=0}^{9N-1} e^{-2\pi i n k/9N} s[k] \qquad (6.38)\\ &= \hat{s}_0(n) + \sum_{\lambda=1}^{4}\left(e^{-2\pi i\lambda n/9N}\hat{s}_\lambda[n] + e^{2\pi i\lambda n/9N}\hat{s}_{-\lambda}[n]\right), \end{aligned}$$

where $\hat{s}_0(n) \stackrel{\text{def}}{=} \sum_{k=0}^{N-1}\omega^{-nk}S_0(k) = \hat{f}_0(n)$ and for $\lambda = \pm1, \pm2, \pm3, \pm4$ there holds

$$\hat{s}_\lambda[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-nk} S_0\left(k+\frac{\lambda}{9}\right) = \sum_{k=0}^{N-1} \omega^{-nk} \frac{1}{N} \sum_{\nu=0}^{N-1} \frac{\hat{f}^0[\nu]}{u^p[\nu]} \zeta^p[\nu] \left(k+\frac{\lambda}{9}\right)$$
$$= \sum_{k=0}^{N-1} \omega^{-nk} \frac{1}{N} \sum_{\nu=0}^{N-1} \omega^{\nu k} \frac{\hat{f}^0[\nu]}{u^p[\nu]} \zeta^p[\nu] \left(\frac{\lambda}{9}\right)$$
$$= \sum_{k=0}^{N-1} \omega^{-nk} \frac{1}{N} \sum_{\nu=0}^{N-1} \omega^{\nu k} \frac{\hat{f}^0[\nu]}{u^p[\nu]} e^{2\pi i \lambda \nu/9N} \tilde{\upsilon^p}_\lambda[n] = \frac{\hat{f}^0[n]}{u^p[n]} e^{2\pi i \lambda n/9N} \tilde{\upsilon}^p_\lambda[n]..$$

The sequence $\tilde{\upsilon}^p_\lambda[n]$ is

$$\tilde{\upsilon}^p_\lambda[n] \stackrel{\text{def}}{=} e^{-2\pi i \lambda n/9N} \zeta^p[n] \left(\frac{\lambda}{9}\right) = \sum_{\nu\in\mathbb{Z}} e^{2\nu\lambda\pi i/9} \left(\frac{\sin \pi\,(n/N+\nu)}{\pi\,(n/N+\nu)}\right)^p. \quad (6.39)$$

By substituting Eq. (6.39) into (6.38), we get

$$\hat{s}[n] = \hat{f}^0[n] \frac{u^p[n] + \sum_{\lambda=1}^4 \left(\tilde{\upsilon}^p_\lambda[n] + \tilde{\upsilon}^p_{-\lambda}[n]\right)}{u^p[n]}$$
$$= \frac{\hat{f}^0[n]}{u^p[n]} \sum_{\nu\in\mathbb{Z}} \left(1 + \sum_{\lambda=1}^4 \left(e^{2\nu\lambda\pi i/9} + e^{-2\nu\lambda\pi i/9}\right)\right) \left(\frac{\sin \pi\,(n/N+\nu)}{\pi\,(n/N+\nu)}\right)^p$$
$$= \frac{\hat{f}^0[n]}{u^p[n]} \sum_{\nu\in\mathbb{Z}} \left(1 + 2\sum_{\lambda=1}^4 \cos\frac{2\nu\lambda\pi}{9}\right) \left(\frac{\sin \pi\,(n/N+\nu)}{\pi\,(n/N+\nu)}\right)^p.$$

It is readily verified that $1 + 2\sum_{\lambda=1}^4 \cos 2\nu\lambda\pi/9 = 9$ when $\nu = 9n$ and zero otherwise. Thus,

$$\hat{s}[n] = \frac{9\hat{f}^0[n]}{u^p[n]} \sum_{\nu\in\mathbb{Z}} \left(\frac{\sin 9\pi(n/9N+\pi\nu)}{9(n/9N+\pi\nu)}\right)^p \quad (6.40)$$
$$= 9^{1-p} \left(1 + 2\cos\frac{2\pi n}{9N}\right)^p \frac{\hat{f}^0[n]}{u^p[n]} \sum_{\nu\in\mathbb{Z}} \left(\frac{\sin 3\pi(n/9N+\pi\nu)}{(n/9N+\pi\nu)}\right)^p$$
$$= 9^{1-p} \left(1 + 2\cos\frac{2\pi n}{9N}\right)^p \left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \frac{\hat{f}^0[n]}{u^p[n]} \sum_{\nu\in\mathbb{Z}} \left(\frac{\sin \pi(n/9N+\pi\nu)}{(n/9N+\pi\nu)}\right)^p$$
$$= 3^{-2p} \left(1 + 2\cos\frac{2\pi n}{9N}\right)^p \left(1 + 2\cos\frac{2\pi n}{3N}\right)^p \tilde{u}^p_2[n] \frac{\hat{f}^0[n]}{u^p[n]}.$$

By comparing Eq. (6.40) with (6.36) and (6.37), we see that $\hat{f}^2[n] = \hat{\mathbf{s}}[n]$, $n = 0, \ldots, 9N-1$ and, consequently, $f^2[k] = S(k/9)$, $k \in \mathbb{Z}$. Repeating the above

reasoning with the initial data set $\mathbf{f}^1$ instead of $\mathbf{f}^0$, we prove that $f^3[k] = S_1(k\,3^{-3}) = S(k\,3^{-3})$, and so on. ■

6.3.5 Practical Implementation

It follows from Theorem 6.2 that, once the samples $\left\{S(k) = f^0[k]\right\},\ k \in \mathbb{Z}$ of a spline $S(t) \in {}^p\mathscr{S}$ are available, the values of the spline at triadic rational points $\{S(k/3^m)\},\ k \in \mathbb{Z},\ m = 1, 2, \ldots,$ are derived by m successive refinement steps with the initial data $\mathbf{f}^0$ and the filters $\mathbf{a}_0^p$, $\mathbf{a}_1^p$,…,$\mathbf{a}_{m-1}^p$, whose frequency responses are given in Eq. (6.34). As in the binary subdivision. this process can be reduced to one DFT, a few multiplications and one IDFT. Due to Eq. (6.34), we get

$$\hat{f}^m[n]_{\tilde{m}} = \frac{\tilde{u}_m^p[n]}{3^{mp}\,u^p[n]}\ \hat{f}^0[n]\ \prod_{l=1}^{m}\left(1 + 2\cos\frac{2\pi n}{\tilde{N}_l}\right)^p. \tag{6.41}$$

Computation of the values $S(k/3^m) = f^m[k]$ of the spline $S(t) \in {}^p\mathscr{S}$ from the samples $S(k) = f^0[k]$ is straightforward:

1. Calculate the $N-$point DFT $\hat{f}^0[n] = \sum_{k=0}^{N-1}\omega^{-kn}\ f^0[k]$ of the initial data $\mathbf{f}^0 = \left\{f^0[k]\right\}$.
2. Derive $\hat{f}^m[n]_{\tilde{m}}$ from Eq. (6.41) bearing in mind that $u^p[n]$ and $\hat{f}^0[n]$ are $N-$periodic sequences, while $\tilde{u}_m^p[n]$, which is calculated according to Proposition 6.1, is $3^m N-$periodic.
3. Implement the $3^m N-$point IDFT

$$S(k/3^m) = f^m[k] = \frac{1}{3^m N}\sum_{n=0}^{3^m N-1} e^{-2\pi ikn/3^m N}\ \hat{f}^m[n]_{\tilde{m}}$$

of the sequence $\hat{\mathbf{f}}^m = \left\{\hat{f}^m[n]_{\tilde{m}}\right\}$.

Calculation of the spline values at triadic rational points is implemented by the MATLAB function `interspliPt.m`.

6.3.5.1 Examples of Odd-Order Splines Generators

Figures 4.1, 4.2, 4.3, 4.4 and 4.5 display generators of the spline spaces ${}^p\mathscr{S}$ where the orders are even, $p = 2, 4, 8, 12$. Those graphs were produced using the binary subdivision. To display the generators of odd-orders spline spaces ${}^p\mathscr{S}$, the ternary subdivision is used. Figure 6.1 displays the interpolaing generators of the spline spaces ${}^p\mathscr{S}$ where the orders are odd, $p = 3, 5, 7, 13$.

Figure 6.2 displays the self-dual generators of the spline spaces ${}^p\mathscr{S}$ where the orders are odd, $p = 3, 5, 7, 13$.

The figures were produced by the MATLAB code `GensplConsP3.m`.

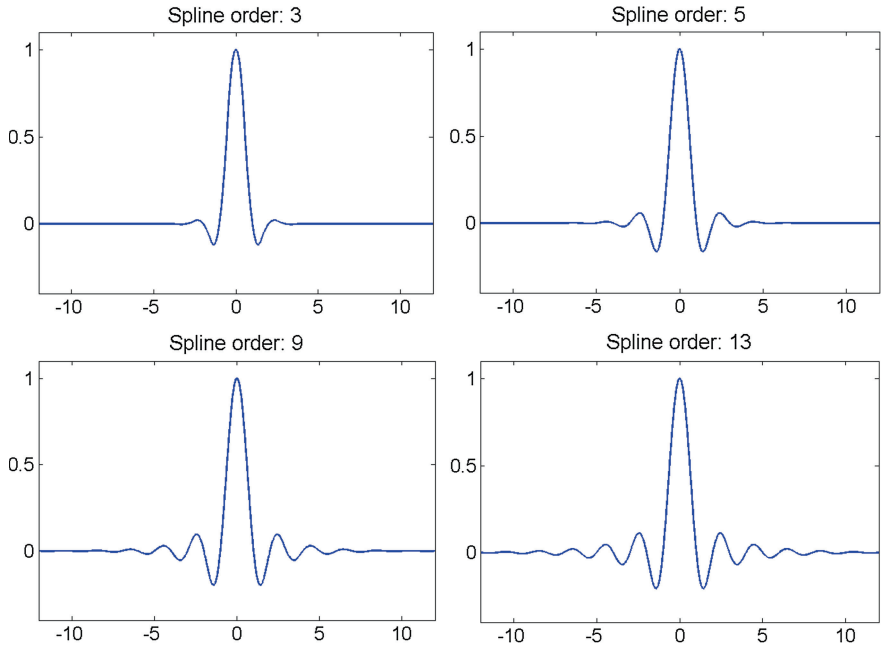


Fig. 6.1 Fundamental splines $L^p(t)$ of odd orders $p = 3, 5, 9$ and 13

6.4 Two-dimensional Spline Subdivision

The subdivision schemes, which provide internal values for 1D splines, are extended to 2D cases in a natural way. A two-dimensional N-periodic spline $S(x, y) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} s[k, n]\, B^p(x - k)\, B^q(y - n)$ from the space ${}^{p,q}\mathscr{S}$ is a 1D spline of order p with respect to x when the variable y is fixed and a 1D spline of order q with respect to y when the variable x is fixed. Therefore, for the grid samples of the spline $\{S(k, n)\}$, $k, n = 0, \ldots, N - 1$, we can first apply a binary (for this $p = 2r$) or a ternary 1D subdivision scheme to the columns of the array $\{S(k, n)\}$ thus producing one of the two arrays

$$\mathbf{s}_m^P = \left\{S(k/P^m, n)\right\},\ k = 0, \ldots, P^m\, N - 1,\ n = 0, \ldots, N - 1,\ P = 2 \text{ or } 3.$$

The next step is an application of a subdivision algorithm to the rows of the array $\mathbf{s}_m^P$, either $P = 2$ or $P = 3$, thus producing one of the following four arrays:

$$\left\{S(k/P^m, n/Q^l)\right\},\ k = 0, \ldots, P^m\, N-1,\ n = 0, \ldots, Q^l\, N-1,\ P = 2 \text{ or } 3,\ Q = 2 \text{ or } 3,$$

and m and l are natural numbers.

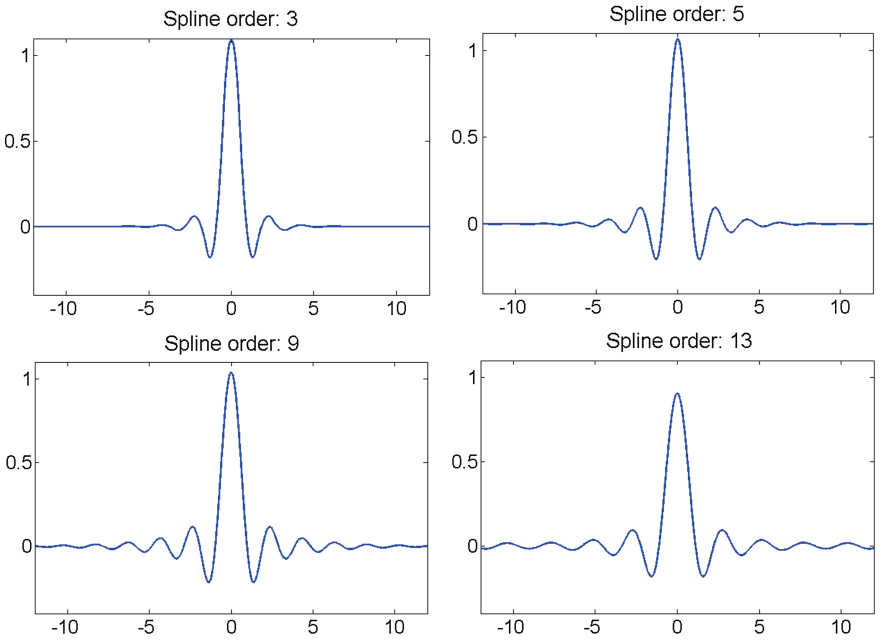


Fig. 6.2 Self-dual splines $\varphi^p(t)$ of odd orders $p = 3, 5, 9$ and 13

We emphasize that a spline $S(x, y)$ can have different orders with respect to x and y. The subdivision algorithms and their depth can be different for the columns and for the rows. In the MATLAB implementation, computations for periodic splines are accelerated because basic operations such as the fast Fourier transforms, multiplications and divisions are applicable to the whole array rather than to separate columns and rows.

Calculation of the 2D spline values at dyadic and triadic rational points is implemented by the MATLAB function `intersplli2DPt.m`.

6.5 Upsampling Signals and Images

An obvious application for 1D and 2D spline subdivision is to upsample discrete-time signals and digital images.

6.5.1 Upsampling Discrete-Time signals

If a signal $\mathbf{x} = \{x[k]\}$, $k = 0, \ldots, N-1$, consists of samples of a smooth periodic function $x[k] = f(k/T)$, the subdivision techniques presented in Sects. 6.2 and 6.3

enable close approximation of the function on the internal points even if the grid $\{k/T\}$ is sparse. When the available samples are affected by noise, then smoothing splines do a good restoration job. If the function $f(t)$ is bandlimited (a trigonometric polynomial), then it is almost perfectly restored by the higher order interpolaing splines.

Example 1: Restoration of a bandlimited signal

The 1-periodic function

$$f(t) \stackrel{\text{def}}{=} \cos 2\pi Ft + 5\sin \pi Ft - 2 * \sin 4\pi Ft - \cos \pi Ft/2, \quad F = 32,$$

is displayed in Fig. 6.3.

The function is sampled on the grid $\{k/256\}$. The signal $\mathbf{f}^0 \stackrel{\text{def}}{=} \{f(k/256)\}$, $k = 0, \ldots, 255$, is used as the initial data. We compared the performances of the subdivision schemes with splines of different orders for restoration of the function $f(t)$ at dyadic and triadic rational points. As expected, the high order splines achieved the best result. The left panel in Fig. 6.4 illustrates the results from the application of six binary subdivision steps with the splines $S^{2r}(t)$ of even orders ranging from 2 to 20, which interpolae the signal $\mathbf{f}^0$. The subdivision produces the splines' $S^{2r}(k/2^{14})$ values. Thus, the signal $\mathbf{f}^0$ is upsampled at the ratio of 1:64. The results are evaluated by the averaged deviations $d^{2r} \stackrel{\text{def}}{=} \sqrt{2^{-14}\sum_{k=0}^{2^{14}-1} |f(k/2^{14}) - S^{2r}(k/2^{14})|}$. The bars in the left panel display the values $D^{2r} \stackrel{\text{def}}{=} \log_{10}(d^{2r})$. The best restoration of the function f is achieved by the 16-th order spline where $D^{16} = -7.39$.

The splines $S^{2r+1}(t)$ of odd orders ranging from 3 to 21 are restored by four steps of the ternary subdivision. The subdivision produces the splines' values $S^{2r+1}(k/K)$, $K = 256 \cdot 3^4 = 20736$. Thus, the signal $\mathbf{f}^0$ is upsampled at the ratio of 1:81. The results are evaluated by the averaged deviations $d^{2r+1} \stackrel{\text{def}}{=} \sqrt{K^{-1}\sum_{k=0}^{K-1} |f(k/K) - S^{2r}(k/K)|}$. The bars in the right panel display the values $D^{2r+1} \stackrel{\text{def}}{=} \log_{10}(d^{2r+1})$. The best restoration result of the function f is achieved by the 17-th order spline where $D^{17} = -7.5$.

The experiments were implemented by the MATLAB code `cos_ups_exam.m`.

Example 2: Restoration of a noised signal

In this example, the original signal is a fragment of the *chirp* function $f(t) = \sin(1/t)$, where $t \in [0.071, 0.971]$. This fragment is displayed in Fig. 6.5. It oscillates with a variable frequency.

In the first experiment the data contains 128 equidistant samples of the function, which are affected by white noise whose STD $= 0.35$. These data were upsampled

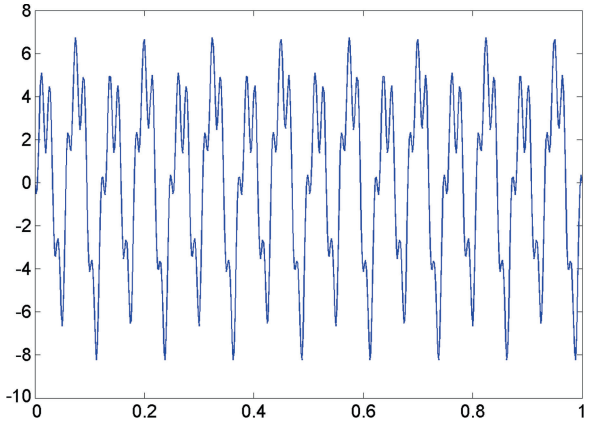


Fig. 6.3 The function $f(t)$

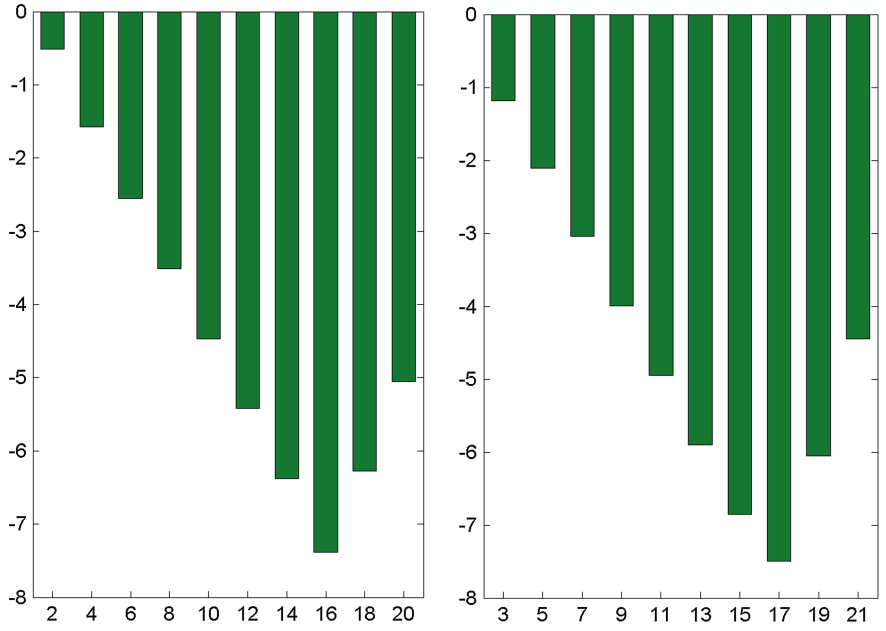


Fig. 6.4 *Left* X-axis—orders of the splines. Y-axis—logarithmic deviations D^{2r} of the splines S^{2r} from the function f (binary subdivision). *Right* X-axis—orders of the splines. Y-axis – logarithmic deviations D^{2r+1} of the splines S^{2r+1} from the function f (ternary subdivision)

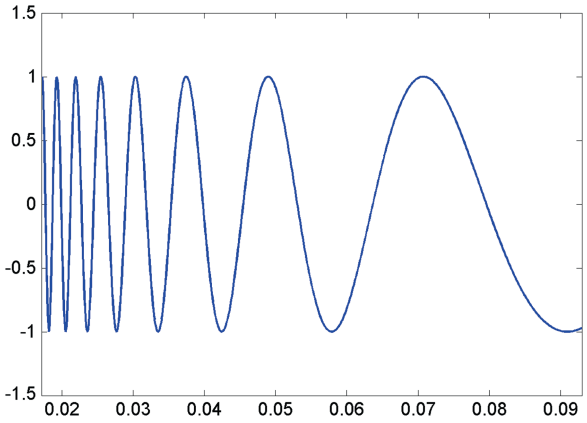


Fig. 6.5 The *chirp* function $f(t) = \sin(1/t)$

by binary subdivision at ratio of 1:8, with the smoothing splines of orders 4, 8 and 12, where the results are displayed in the top, middle and bottom of the left side of Fig. 6.6, respectively. To be specific, the smoothing splines were constructed on the grid $\{k\}$, $k = 1, \ldots, 128$, and their values at the points $\{l/8\}$, $k = 1, \ldots, 1024$, were computed by binary subdivision.

In the second experiment, the data was decimated by factor of 2, thus, the initial data consisted of 64 noised samples. These data were upsampled at the ratio of 1:16 by the application of subdivision with the smoothing splines of orders 4, 8 and 12. The experiments were implemented by the MATLAB code `smoothing_example1_sin.m`. The results are displayed in the right side of Fig. 6.6. One can observe that, in spite of the presence of strong noise and low-rate sampling, the original signal was satisfactorily restored. Note that order 4 (cubic) splines best restore the low-frequency part of the signal, while the splines of the 12th order restore the high-frequency region well.

6.5.2 Upsampling Digital Images

Upsampling an image using 2D interpolaing splines increases its resolution. For this, the image pixels are treated as the grid samples of a 2D spline and the initial data array is upsampled by the spline's values in internal points, which are derived by application of binary or ternary subdivision as it is described in Sects. 6.2 and 6.3, respectively. The 2D subdivision is outlined in Sect. 6.4. When pixels of the image are affected by noise, the smoothing splines provide a good approximation.

Performance of the spline subdivision schemes is illustrated in a few experiments with the benchmark images "Lena" and "Barbara", which are presented by the 512×512 pixels arrays denoted by $\mathbf{L}$ and $\mathbf{B}$, respectively.

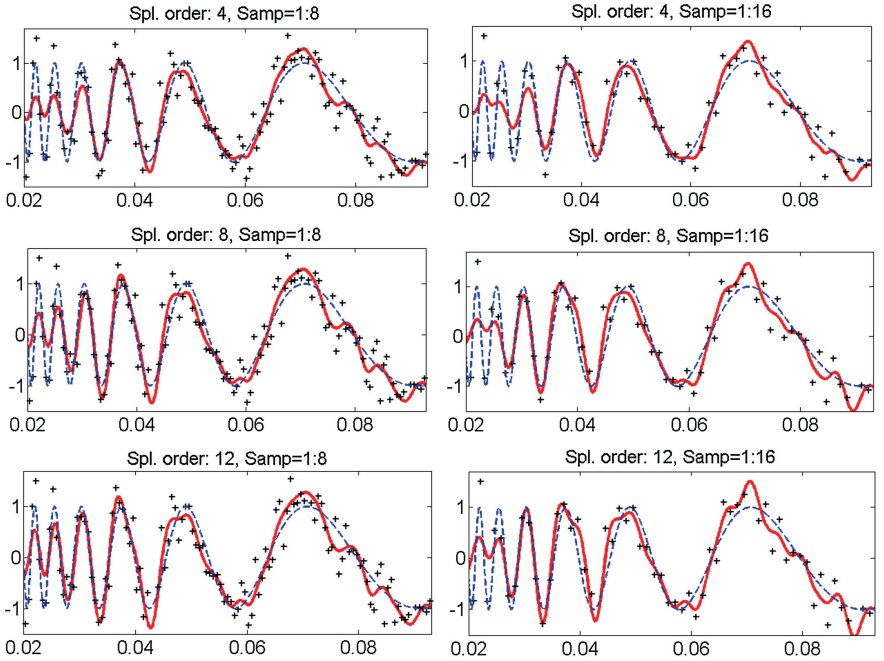


Fig. 6.6 *Left* restoration of the function $f(t) = \sin(1/t)$ from 128 noised samples by the smoothing splines of order 4 (*top*), 8 (*middle*) and 12 (*bottom*). *Right* restoration from 64 samples. *Dotted line* indicates the original function, "pluses"—available data, and *solid lines*—restoring splines

Example 1: Approximation of the "Lena" image from decimated arrays

In this example, the "Lena" image is restored after it was decimated by ratio of 2:1 and of 4:1 to generate the data arrays $\mathbf{d}_2$ of size 256×256 and $\mathbf{d}_4$ of size 128×128. The low-pass anti-aliasing filtering was not applied to the decimated arrays. The data arrays $\mathbf{d}_s$, $s = 2, 4$ were interpolaed by the even-order splines $S_s^{2r}(x, y)$, $s = 2, 4$, that is $S_s^{2r}(k, n) = d_s[k, n]$. Then, the original array was approximated by the splines values $L(k, n) \approx S_2^{2r}(k/2, n/2)$ and $L(k, n) \approx S_4^{2r}(k/4, n/4)$, $k, n, = 0, \ldots, 511$.

The proximity between an approximated image $\tilde{\mathbf{L}}$ and the original image $\mathbf{L}$ is evaluated visually and by the Peak-Signal-to-Noise ratio (PSNR)

$$PSNR \stackrel{\text{def}}{=} 10 \log_{10} \left(\frac{M\, 255^2}{\sum_{k=1}^{M} (x_k - \tilde{x}_k)^2} \right) dB \tag{6.42}$$

where M is the number of pixels in the image (in our experiments, $M = 512^2$), $\{x_k\}_{k=1}^{M}$ are the original pixels of the image $\mathbf{L}$, and $\{\tilde{x}_k\}_{k=1}^{M}$ are the pixels of the image $\tilde{\mathbf{L}}$.

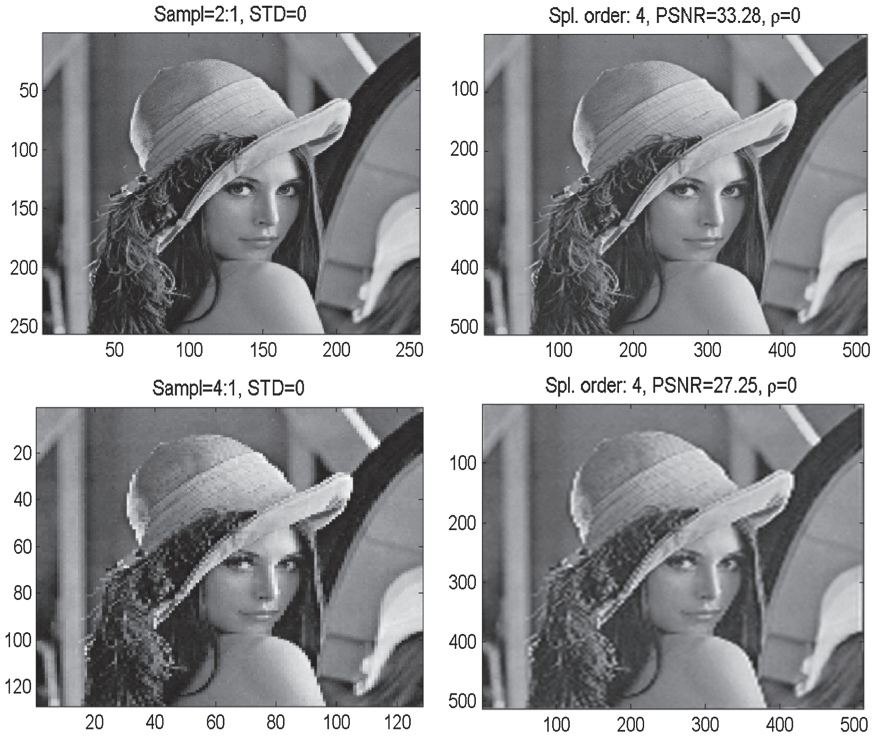


Fig. 6.7 *Left* Decimated data arrays. *Top* $\mathbf{d}_2$, bottom $\mathbf{d}_4$. *Right* the "Lena" image approximated by the interpolaing cubic splines subdivision from $\mathbf{d}_2$ (*top*) and $\mathbf{d}_4$ (*bottom*)

It turns out that the best PSNR was achieved by using subdivision with the cubic splines. Figure 6.7 displays the results of this approximation. We observe that the approximation from the array $\mathbf{d}_2$ is quite satisfactory (PSNR $= 33.28$), while the approximation from the array $\mathbf{d}_4$ is blurred (PSNR $= 27.25$).

Example 2: Restoration of the "Lena" image from noised decimated arrays

In this example, the "Lena" image was decimated by factors of 2:1 and 4:1 to generate the data arrays $\mathbf{d}_2$ of size 256×256 and $\mathbf{d}_4$ of size 128×128, respectively. Then, the arrays were corrupted by a moderate Gaussian noise whose $STD = 10$. The data arrays $\mathbf{d}_s, s = 2, 4$ were approximated by the even-order smoothing splines $S^{2r}_{\rho,s}(x, y),\ s = 2, 4$, such that $S^{2r}_{\rho,s}(k, n) \approx d_s[k, n]$, where the optimal values of the parameter ρ were derived from Eq. (5.14). Then, the original array was restored by the splines values $L(k, n) \approx S^{2r}_{\rho,2}(k/2, n/2)$ and $L(k, n) \approx S^{2r}_{\rho,4}(k/4, n/4)$, $k, n, = 0, \ldots, 511$. As before, the best PSNR was produced by the subdivision with cubic splines. Figure 6.8 displays the results of this restoration. We observe that

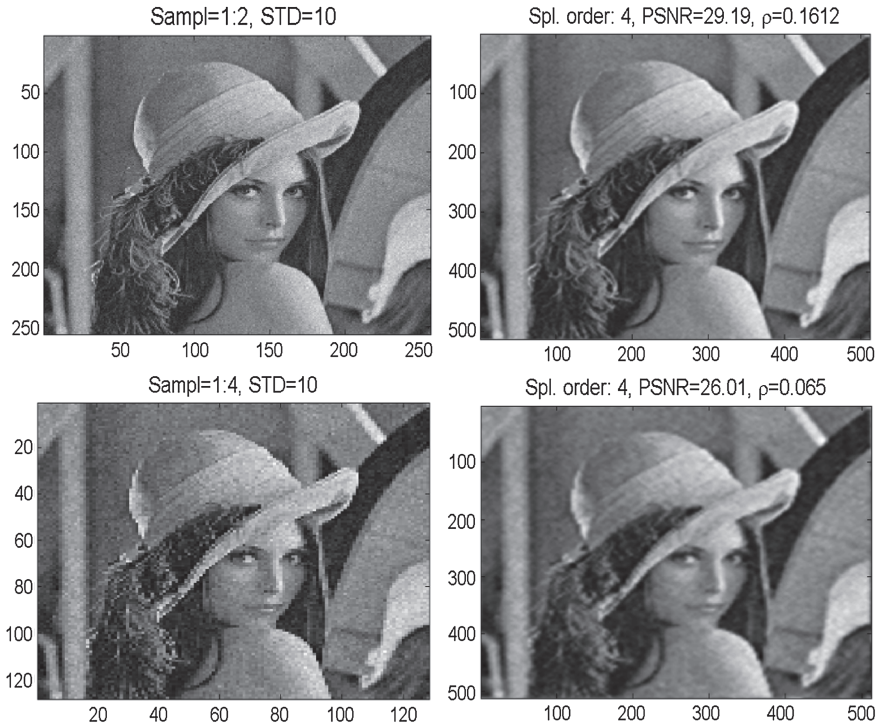


Fig. 6.8 *Left* Decimated *noised* images. *Top* $\mathbf{d}_2$, *bottom* $\mathbf{d}_4$. *Right* the "Lena" image restored by the binary subdivision using the cubic smoothing splines from $\mathbf{d}_2$ (*top*) and from $\mathbf{d}_4$ (*bottom*)

noise is suppressed in the restored images. Restoration from the array $\mathbf{d}_2$ produces a satisfactory quality ($PSNR = 29.19$), while the restoration from the array $\mathbf{d}_4$ is blurred ($PSNR = 26.01$).

Experiments in Example 1 and 2 are implemented by the MATLAB code `smoothing_restoration2.m`. Comparison of the performance of different splines is done by the MATLAB code `smoothing_restoration2_mult.m`.

Example 3: Upsampling the "Barbara" image

In this example, we upsample the "Barbara" image of size 512×512. It is given by the array $\mathbf{B}$ of pixels. We used the 2D spline $S^{7,8}(x, y)$, which interpolaes the initial data, that is $S^{7,8}(k, n) = B[k, n]$. The spline $S^{7,8}(x, y)$ is of order 7 in the vertical direction and order 8 in the horizontal direction. By application of 3 steps of the ternary subdivision in the vertical direction and 4 steps of the binary subdivision in the horizontal direction, we generate the array $\mathbf{S} \stackrel{\text{def}}{=} \left\{S^{7,8}(k/27, n/16)\right\}$, where $k = 0, \ldots, 13823$ and $n = 0, \ldots, 8191$. The array $\mathbf{S}$ is used for the upsampled image.

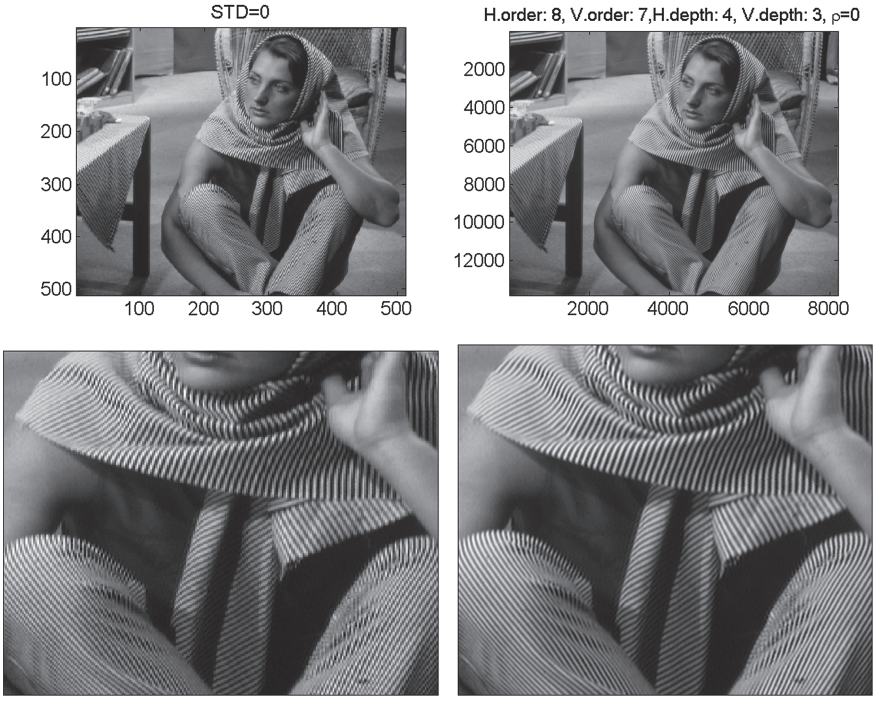


Fig. 6.9 *Left* The original "Barbara" image in two formats. *Top* The whole image. *Bottom* A fragment of the image. *Right* The image upsampled after the application of the interpolaing splines $S^{7,8}(x, y)$ using ternary subdivision in the *vertical direction* and binary subdivision in the *horizontal direction*

The result is displayed in Fig. 6.9. We see that upsampling significantly increased the image resolution.

Example 4: Upsampling the "Barbara" image that was corrupted by noise

In this example, the pixels array $\mathbf{B}$ is corrupted by moderate Gaussian noise with STD=10. The image is upsampled by using the 2D bicubic smoothing spline $S_\rho^{4,4}(x, y)$, which approximates the original noised data, that is, $S_\rho^{4,4}(k, n) \approx B[k, n]$. The optimal value of the parameter ρ was derived from Eq. (5.14). By application of 2 steps of the ternary subdivision in either direction, the array $\mathbf{S}_\rho \stackrel{\text{def}}{=} \left\{S_\rho^{4,4}(k/9, n/9)\right\}$, $k, n = 0, \ldots, 4607$, is generated. The array $\mathbf{S}_\rho$ is used as the upsampled image. The upsampling result is displayed in Fig. 6.10. We observe that the upsampling suppressed the noise and increased the image resolution. Experiments in Example 3 and 4 are implemented by the MATLAB code `smoothing_upsamp2_ex.m`.

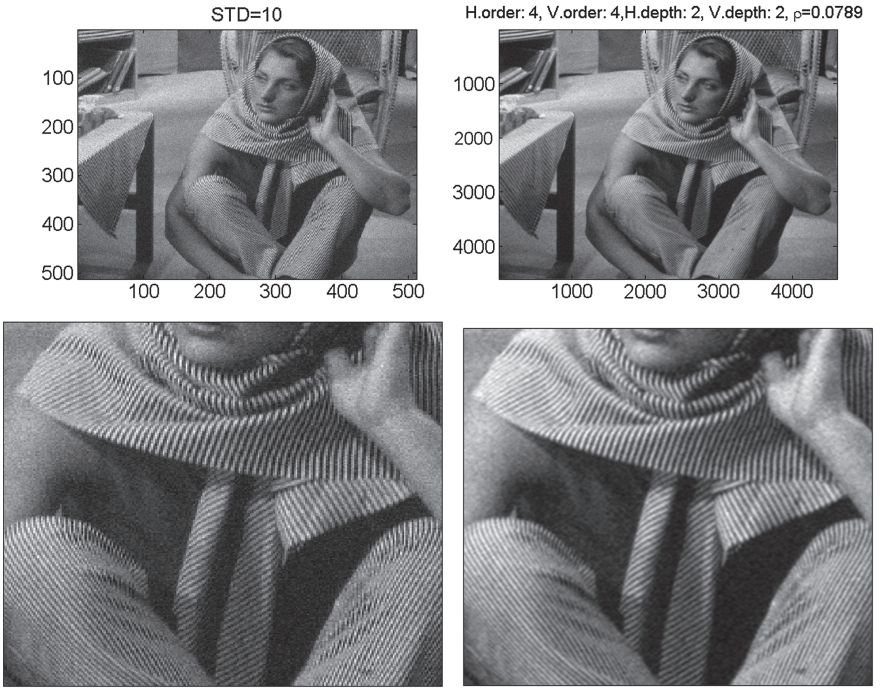


Fig. 6.10 *Left* The noised "Barbara" image in two formats. *Top* The original image. *Bottom* A fragment from the image. *Right* The image upsampled by the smoothing splines $S_\rho^{4,4}(x, y)$ using the ternary subdivision in each direction

Comments

The above examples demonstrate that the parameterized smoothing splines, where the regularization parameter ρ is derived automatically from Eqs. (5.7) and (5.14), efficiently suppress broad-band noise. Their mode of operation on non-decimated signals and images consists in adaptive low-pass filtering the input data. The pass-band of the filters, which is determined by the parameter ρ, depends on signal-to-noise ratio in available data. Suppression of strong noise is achieved by narrowing the pass-band. A side effect is that fine details of the objects become blurred. A possible way to retain fine details in the image while suppressing noise is application of wavelet transforms followed by adaptive thresholding of the transform coefficients [1–3]. Another approach is to use different values of ρ for different frequency domains of images. This approach, which is based on utilizing spline wavelet packets will be discussed in Chap. 12.

An advantage of the smoothing splines is the simplicity of the implementation and automatic adaptation to the available data. Processing is fast due to the utilization of the fast Fourier transforms.

References

1. S.G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression. IEEE Trans. Image Process. **9**(9), 1532–1546 (2000)
2. D. Donoho, I. Johnstone, Ideal spatial adaptation via wavelet shrinkage. Biometrika **81**(3), 425–455 (1994)
3. D. Donoho, I. Johnstone, Adapting to unknown smoothness via wavelet shrinkage. J. Amer. Stat. Assoc. **90**(432), 1200–1224 (1995)
4. N. Dyn, Analysis of convergence and smoothness by the formalism of Laurent polynomials, in *Tutorials on Multiresolution in Geometric Modelling*, ed. by A. Iske, E. Quak, M.S. Floater (Springer, Berlin, 2002), pp. 51–68
5. N. Dyn, J.A. Gregory, D. Levin, Analysis of uniform binary subdivision schemes for curve design. Constr. Approx. **7**(1), 127–147 (1991)
6. V. Zheludev, Interpolatory subdivision schemes with infinite masks originated from splines. Adv. Comput. Math. **25**(4), 475–506 (2006)

Chapter 7
Spline Algorithms for Deconvolution and Inversion of Heat Equation

Abstract In this chapter, we present algorithms based on Tikhonov regularization for solving two related problems: deconvolution and inversion of heat equation. The algorithms, which utilize the SHA technique, provide explicit solutions to the problems in one and two dimensions.

7.1 Spline Algorithms for Deconvolution

The deconvolution problem is related to the denoising problem. The difference is that, prior to corruption by noise, the input data has been subjected to blurring. This effect can originate from measuring a signal by an instrument, transmission through a channel, propagation of a seismic signal through the earth, and observation of a distant or small object through an optical device, to name a few. In many cases, the result of blurring can be modeled as convolution of the original signal or image f with some kernel h, which is effectively bandlimited:

$$g(t) = \int_{-\infty}^{\infty} h(t-\tau) f(\tau)\, d\tau, \tag{7.1}$$

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x-\xi, y-\eta)\, h(\xi, \eta)\, d\xi\, d\eta. \tag{7.2}$$

Deconvolution is a required operation in many signal processing applications such as system identification, spectroscopy, seismic processing, image deblurring, acoustic and audio processing and so on. Due to bandlimitedness of the kernels, the high-frequency components of the input object f become suppressed and their presence in the output g is minimal, if any. Typically, however, errors, which constitute a broad-band noise, present in measurements of the output g. A straightforward inversion of the convolution operator leads to an amplification of these errors. Therefore,

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_7,

this inversion is far from being stable and robust. In that sense, the deconvolution problem is regarded as an ill-posed one. An additional complication stems from the fact that, in practice, the measurements of the convolved signal (image) are available in a discrete set of points. Under such circumstances, an exact restoration of the input object is not possible, and a sound approach, which was formulated in the classical work by Wiener [7], consists in finding a reasonable compromise between approximation of the available data and stability of the solution. By now, a number of efficient deconvolution algorithms are based on the Tikhonov regularization method [5, 6].

The Tikhonov methodology, in which approximate solution is derived as a minimizer of a parameterized quadratic functional, is, to some extent, similar to construction of smoothing splines. Even more similarity arises when the output objects g are presented by sets of discrete noised measurements and the solutions f are approximated by splines, which map the discrete data into spaces of smooth functions. An additional advantage of using splines for deconvolution application lies in their abilities to effectively control the smoothness of the solution.

In this section, spline-based algorithms for solving the convolution integral equations are described. The algorithms are based on Tikhonov regularization. As in the smoothing splines construction, the working tool is the SHA, which provides the approximate solutions in an explicit form together with a unified computational scheme, which is based on the fast Fourier transform (FFT).

7.1.1 Solution of One-Dimensional Convolution Equations

In the one-dimensional convolution Eq. (7.1) $f(t)$ is an unknown sought-after function, $h(t)$ is the kernel and $g(t)$ is the given data function. Typically in applications, the functions $h(t)$ and the unknown function $f(t)$ are compactly supported: thus, necessarily, $g(t)$ has a compact support as well. Then, the deconvolution problem can be reduced to finding a T-periodic solution of the equation

$$g(t) = h(t) \circledast f(t) = \int_0^T h(t-\tau) f(\tau)\, d\tau, \tag{7.3}$$

where the unknown function $f(t)$, the blurring kernel $h(t)$ and the data function $g(t)$ are assumed to be T-periodic. If $h(t) \in C^l$ and $f(t) \in C^m$ then $g(t) \in C^{l+m+1}$.

A theoretical solution of the Eq. (7.3) is straightforward. Due to Eq. (1.9), the Fourier coefficients of the output signal $c_n(g) = c_n(f)\, c_n(h)$. If all the Fourier coefficients of the kernel $c_n(h)$ are non-zero then the Fourier coefficients of the solution $c_n(f) = c_n(g)/c_n(h)$ and

$$f(t) = \frac{1}{T} \sum_{n \in \mathbb{Z}} \frac{c_n(g)}{c_n(h)}\, e^{2\pi i\, nt/T}. \tag{7.4}$$

The Fourier coefficients of the kernel $c_n(h)$ tend to zero as $n \to \infty$, but the coefficients $c_n(g)$ tend to zero even faster. Thus, theoretically, the solution given in Eq. (7.4) is stable. However, in real life, the output signal $g(t)$ is known up to some errors: $y(t) = g(t) + e(t)$. Generally, there is no reason to assume that the coefficients $c_n(e)$ tend to zero according to the decay of $c_n(h)$ (if they tend to zero at all). Therefore, application of the direct inversion method from Eq. (7.4) to the available signal $y(t)$

$$\tilde{f}(t) = \frac{1}{T}\sum_{n\in\mathbb{Z}} \frac{c_n(y)}{c_n(h)}\, e^{2\pi i\, nt/T} = \frac{1}{T}\sum_{n\in\mathbb{Z}} \frac{c_n(g)}{c_n(h)}\, e^{2\pi i\, nt/T} + \frac{1}{T}\sum_{n\in\mathbb{Z}} \frac{c_n(e)}{c_n(h)}\, e^{2\pi i\, nt/T}$$

results in an unstable solution. However small is the magnitude of the error signal $|e(t)|$, the signal $\tilde{f}(t)$ can comprise strong high-frequency components, which do not exist in the original signal $f(t)$. Therefore, a regularization, which provides a stability to the solution at the expense of some deviation from the available data signal $y(t)$, is needed.

In most practical situations, the available data is being sampled on the grid $\{t_k\}$. These samples are corrupted by random noise. We assume that the grid is uniform $\{t_k = Tk/N\}$. Let $N = 2^j$ ($j \in \mathbb{N}$) be the number of nodes on the grid. For simplicity, we assume that $T = N$. Denote $\mathbf{g} = \{g(k)\}_{k=0}^{N-1}$, $\mathbf{h} = \{h(k)\}_{k=0}^{N-1}$ and $\mathbf{y} = \mathbf{g} + \mathbf{e}$, where $\mathbf{e} = \{e_k\}_{k=0}^{N-1}$ is the vector of random errors, which we assume to be a zero-mean white noise.

7.1.1.1 Parameterized Spline Solution

An approximate solution to the Eq. (7.3) will be obtained as an even-order spline from the space ${}^p\tilde{\mathscr{S}}$, where $p = 2r > m$. Thus, splines from ${}^p\tilde{\mathscr{S}}$ belong to the space C^m. Given a sampled data array $\mathbf{y} = \{y[k]\}$ and a kernel data array $\mathbf{h} = \{h(k)\}$, we construct the interpolatory spline

$$\Upsilon(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \mu[n]\, \zeta^q[n](t) \in {}^q\tilde{\mathscr{S}}, \quad \mu[n] = \frac{\hat{h}[n]}{u^q[n]}, \tag{7.5}$$

which interpolates the kernel $h(t)$ on the grid $\{k\}$.

The further design is a generalization of the design of smoothing splines described in Sect. 5.1. The solution is approximated by the spline $S_\rho(t) \in {}^{2r}\mathscr{S}$, which minimizes the functional $J_\rho(S) \stackrel{\text{def}}{=} \rho I(S) + E(S)$, where

$$I(S) \stackrel{\text{def}}{=} \|S^{(r)}\|^2, \quad E(S) \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} (\Upsilon \circledast S(k) - y[k])^2$$

and ρ is a numerical parameter. Let a spline $S(t) \in {}^{2r}\mathscr{S}$ be represented as

$$S(t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta^{2r}[n](t).$$

Then, similarly to Eq. (5.3),

$$I(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} w^{2r}[n]\, u^{2r}[n]\, |\xi[n]|^2, \quad w^{2r}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^{2r}.$$

Combining Eqs. (4.33) and (7.5) provides the expression for the sampled circular convolution

$$\Upsilon \circledast S(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, u^{2r+q}[n]\, \xi[n]\, \mu[n].$$

The signal $\mathbf{y}$ is expressed via its DFT:

$$y[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \hat{y}[n] \omega^{kn}, \quad \hat{y}[n] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn} y[k].$$

Hence, by using the Parseval identity (1.24), the discrepancy functional is presented as

$$E(S) = \sum_{k=0}^{N-1} (\Upsilon \circledast S(k) - y[k])^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left| u^{2r+q}[n]\, \xi[n]\, \mu[n] - \hat{y}[n] \right|^2.$$

Consequently,

$$J_\rho(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \rho |\xi[n]|^2\, w^{2r}[n] + \left| u^{2r+q}[n]\, \xi[n]\, \mu[n] - \hat{y}[n] \right|^2.$$

Similarly to Proposition 5.3, the following fact is derived.

Proposition 7.1 *Minimum of the functional $J_\rho(S)$ on the spline space ${}^{2r}\mathscr{S}$ is achieved by the spline*

$$S_\rho[\mathbf{y}, \mathbf{h}](t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi_\rho[n]\, \zeta^{2r}[n](t), \quad \xi_\rho[n] = \frac{u^{2r+q}\, \mu^*[n]\, \hat{y}[n]}{A_\rho[n]}$$

$$A_\rho[n] \stackrel{\text{def}}{=} \rho\, w^{2r}[n]\, u^{2r}[n] + (|\mu[n]|\, u^{2r+q}[n])^2, \quad w^{2r}[n] = \left(2 \sin \frac{\pi n}{N}\right)^{2r} \quad (7.6)$$

Equations (4.30) and (7.5) imply that the values of the spline at grid points

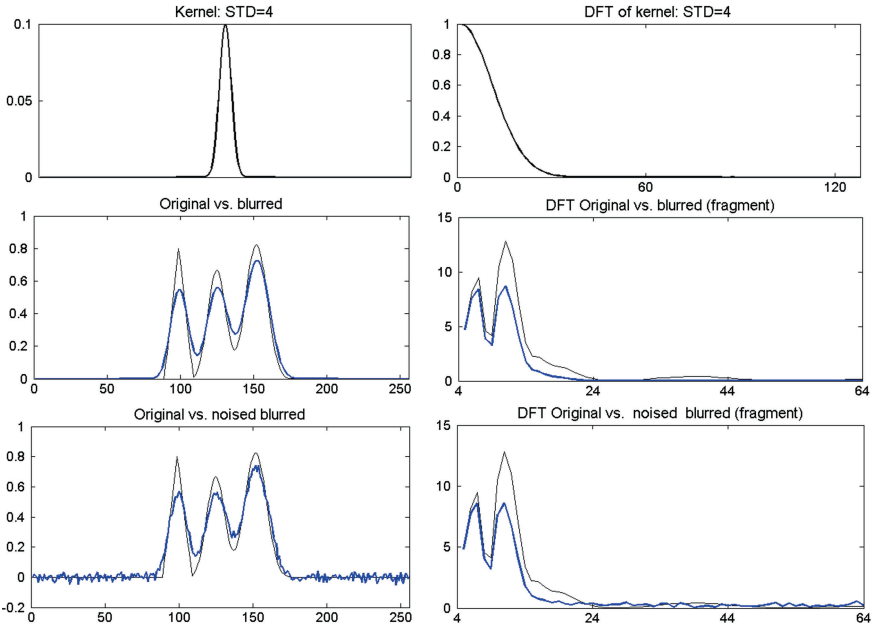


Fig. 7.1 *Top left* the blurring kernel (Gaussian curve STD $\sigma = 4$). *Top right* its DFT. *Center left* the original signal (*thin line*) versus the blurred signal (*thick line*). *Center right* fragments of DFT of the original signal (*thin line*) and the blurred signal (*thick line*). *Bottom left* the original signal (*thin line*) versus blurred noised signal (*thick line*). *Bottom right* fragments of DFT of the original signal (*thin line*) and the blurred noised signal (*thick line*)

$$S_\rho[\mathbf{y},\mathbf{h}](k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \xi_\rho[n]\, u^{2r}[n]\, \omega^{kn}$$
$$= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{u^{2r}[n]\, u^{2r+q}\, \mu^*[n]\, \hat{y}[n]}{A_\rho[n]}. \tag{7.7}$$

It is seen from Eq. (7.7) that computation of the samples of $\mathbf{s}_\rho = \left\{S_\rho[\mathbf{y},\mathbf{h}](k)\right\}$, $k = 0, ..., N-1$, reduces to circular convolution $\mathbf{s}_\rho = \mathbf{d}_\rho \circledast \mathbf{y}$ of the discrete-time signal $\mathbf{y} = \{y[k] = g(k) + e_k\}$, $k = 0, ..., N-1$, with the signal $\mathbf{d}_\rho = \left\{d_\rho[k]\right\}$, whose DFT is

$$\hat{d}_\rho[n] = \frac{u^{2r}[n]\, u^{2r+q}\, \mu^*[n]}{\rho\, u^{2r}[n]\, (2\,\sin \pi n/N)^{2r} + (|\mu[n]|\, u^{2r+q}[n])^2},$$
$$\hat{d}_0[n] = \frac{u^{2r}[n]\, u^q[n]}{u^{2r+q}[n]\, \hat{h}[n]}. \tag{7.8}$$

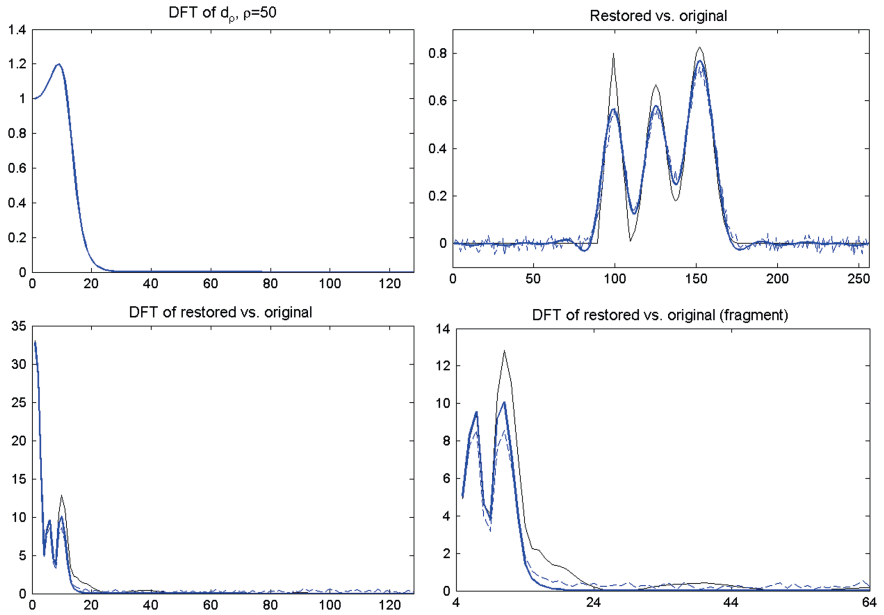


Fig. 7.2 *Top left* magnitude response of the filter $\mathbf{d}_\rho$. *Top right* the original signal $\mathbf{f}$ (*thin line*) versus the input signal $\mathbf{y}$ (*dashed line*) and the restored signal $\mathbf{s}_\rho$, $\rho = 50$, (*thick line*). *Bottom left* DFT of $\mathbf{f}$ (*thin line*) versus DFT of $\mathbf{y}$ (*dashed line*) and of $\mathbf{s}_\rho$ (thick line). *Bottom right* fragments of those DFT

Design of the deconvolution spline is implemented by the MATLAB function `convsplaP.m`.

High-frequency components of the kernel $h(t)$ are close or equal to zero. Consequently, $\mu[n] = \hat{h}[n]/u^q[n]$ and the denominator in $\hat{d}_0[n]$ are close or equal to zero when n approaches $N/2$. Thus, the circular convolution of $\mathbf{d}_0 \circledast \mathbf{y}$, which actually is the direct inversion of the convolution operator, is unstable. The term $\rho\, u^{2r}[n]\ (2\ \sin \pi n/N)^{2r}$ serves as a stabilizer. It prevents the high-frequency decay of the denominator in Eq. (7.7). When n is small, $\hat{d}_\rho[n] \approx \hat{d}_0[n]$. Thus, the lower frequencies remain practically undisturbed if the regularization parameter ρ is not too big. In addition, presence of the factor $\mu^*[n]$ in the numerator of $\hat{d}_\rho[n]$ suppresses the high-frequency components of the noise.

Once the sampling $\mathbf{s}_\rho$ is calculated, the spline $S_\rho[\mathbf{y}, \mathbf{h}](t)$ is constructed as a spline, which interpolates the signal $\mathbf{s}_\rho$.

The influence of the parameter choice on the signal's restoration is illustrated in the following figures. The upper frames in Fig. 7.1 display the blurring kernel, which is the Gaussian curve $h(t) = (2\pi\sqrt{\sigma})^{-1}\, e^{-(t/\sqrt{2}\sigma)^2}$, $\sigma = 4$ and the DFT of its sampled version. Obviously, the kernel is effectively bandlimited. The central left frame displays the original signal $f(t)$ (thin line) versus the blurred signal $g(t)$ (thick line). The right frame displays fragments of the DFT $\hat{f}[n]$ (thin line) and $\hat{g}[n]$ (thick

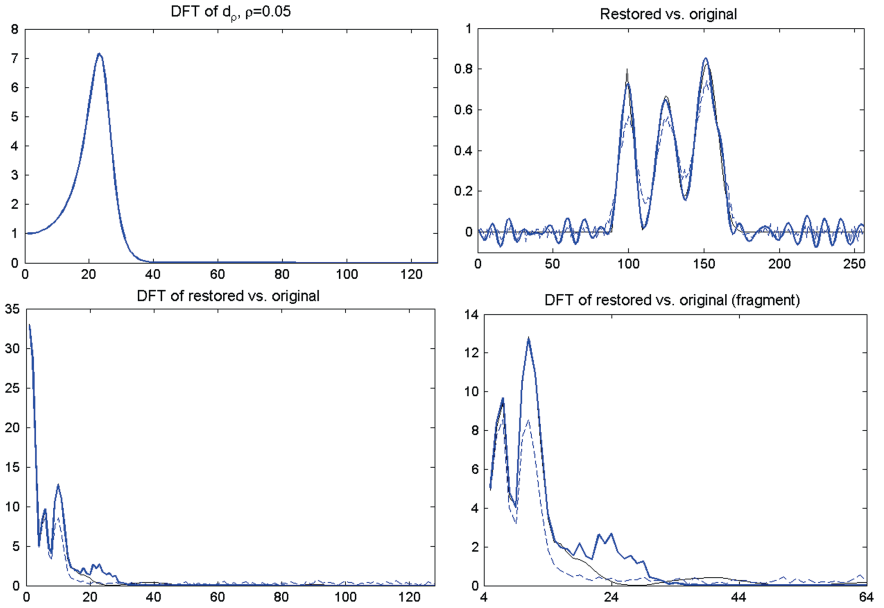


Fig. 7.3 *Top left* magnitude response of the filter $\mathbf{d}_\rho$. *Top right* the original signal $\mathbf{f}$ (*thin line*) versus the input signal $\mathbf{y}$ (*dashed line*) and the restored signal $\mathbf{s}_\rho$, $\rho = 0.05$, (*thick line*). *Bottom left* DFT of $\mathbf{f}$ (*thin line*) versus DFT of $\mathbf{y}$ (*dashed line*) and of $\mathbf{s}_\rho$ (thick line). *Bottom right* fragments of those DFT

line) of sampled functions f and g. The latter is suppressed compared to the former. The bottom frames display the sampled signal $\mathbf{f}$ and its DFT (thin lines) versus the blurred noised signal $\mathbf{y} = \{g(k) + e_k\}$ and its DFT $\hat{y}[n]$ (thick lines). We observe that middle frequencies in $\hat{y}[n]$ are significantly suppressed compared to $\hat{f}[n]$ while there exist erroneous high-frequency components.

Figures 7.2 and 7.3 illustrate reconstruction of the function $f(t)$, which is depicted in Fig. 7.1, from the blurred noised data $\mathbf{y} = \{g(k) + e_k\}$ (bottom left frame in Fig. 7.1). The function is restored by the cubic spline $\mathbf{s}_\rho = \left\{S_\rho[\mathbf{y}, \mathbf{h}](k)\right\}, k = 0, ..., N-1$ by the circular convolution $\mathbf{s}_\rho = \mathbf{d}_\rho \circledast \mathbf{y}$, where the signal $\mathbf{d}_\rho$ is defined in Eq. (7.8). In both figures, the upper left frames display the DFT of the signals $\mathbf{d}_\rho$, while the upper right frames display the signal $\mathbf{f}$ versus the input signal $\mathbf{y}$ and the restored signal $\mathbf{s}_\rho$. The bottom frames display the DFT of these three objects.

Figure 7.2 demonstrates the results of the experiment with a large value of the regularization parameter $\rho = 50$. We observe in the upper left frame that the passband of the filter signal $\mathbf{d}_\rho$ is very narrow and the magnitudes of the DFT do not exceed 1.2. As a result, the higher frequencies in the signal $\mathbf{y}$ become suppressed while middle frequencies are almost intact. Thus, the high-frequency noise in the output signal $\mathbf{s}_\rho$ is almost completely eliminated but the essential structure of $\mathbf{f}$ remains oversmoothed.

Figure 7.3 illustrates another extreme case when $\rho = 0.05$ is too small. Now the passband of the filter signal $\mathbf{d}_\rho$ is wider and the magnitudes of the DFT have a sharp upsurge at the middle frequencies. As a result, the middle frequencies in the output signal $\mathbf{s}_\rho$ arrive significantly elevated over the respective frequencies of the signal $\mathbf{f}$. This leads to strong parasite oscillations in $\mathbf{s}_\rho$, although the structure of $\mathbf{f}$ is resolved perfectly. The plots were produced by the MATLAB code `regExampleP.m`.

Thus, a procedure to derive a proper value of ρ is needed.

7.1.1.2 Optimal Regularization Parameter

This procedure is similar to the search for the optimal parameter of a smoothing spline. Assume we are able to estimate the squared norm $\varepsilon^2 = \sum_{k=0}^{N-1} e_k^2$ of the error vector. The regularization parameter ρ is derived from the solution of the following problem.

Problem ORP
Among the splines $S_\rho[\mathbf{y}, \mathbf{h}](t)$ *defined in* Eq. (7.6), *find a spline that minimizes the functional* $I(S_\rho)[\mathbf{y}, \mathbf{h}]$ *under the constraint* $E(S_\rho[\mathbf{y}, \mathbf{h}]) \le \varepsilon^2$.

Loosely speaking, we are looking for the smoothest spline, among the splines $S_\rho[\mathbf{y}, \mathbf{h}]$, for which the standard deviation of the vector $\{\Upsilon \circledast S_\rho[\mathbf{y}, \mathbf{h}](k)\}$ from the data vector $\mathbf{y}$ does not exceed the standard deviation of the vector $\mathbf{y}$ from the exact data vector $\mathbf{g} = \{g(k)\}$.

Denote $e(\rho) \stackrel{\text{def}}{=} E(S_\rho[\mathbf{y}, \mathbf{h}])$. It can happen that some coordinates $\mu[n]$ of the interpolatory spline Υ are zero. Then, denote by ς the set of indices for which $\mu[n] = 0$. If ς is not empty then denote

$$\theta(y) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n \in \varsigma} |\hat{y}[n]|^2.$$

It follows from Eq. (7.6) that

$$e(\rho) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left(\frac{\rho\, u^{2r}[n]\, w^{2r}[n]\, \hat{y}[n]}{A_\rho[n]} \right)^2.$$

The function $e(\rho)$ grows strictly monotonically while $e(0) = \theta(y)$ and $\lim_{\rho \to \infty} e(\rho) = 1/N \sum_{n \in \varsigma} |\hat{y}(n)|^2 = \|\mathbf{y}\|^2$.

On the other hand, the function

$$i(\rho) \stackrel{\text{def}}{=} I(S_\rho[\mathbf{y}, \mathbf{h}]) = \frac{1}{N} \sum_{n \notin \varsigma} \frac{u^{2r}[n]\, w^{2r}[n]\, |u^{2r+q}[n]\, \hat{h}[n]\, \hat{y}[n]|^2}{\left(\rho\, u^{2r}[n]\, w^{2r}[n] + (|\hat{h}[n]|\, u^{2r+q}[n])^2 \right)^2}$$

decays strictly monotonically while $\lim_{\rho \to \infty} i(\rho) = 0$.

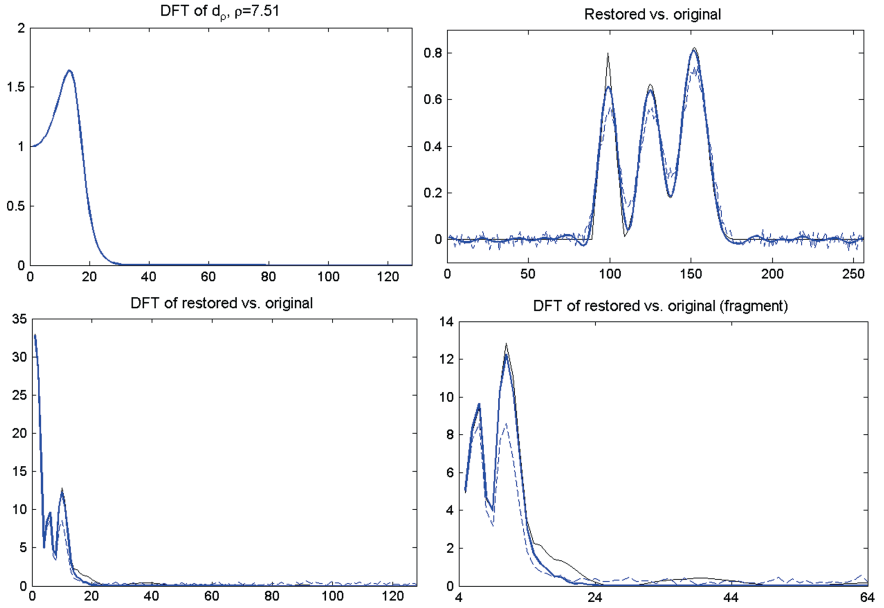


Fig. 7.4 *Top left* magnitude response of the filter $\mathbf{d}_\rho$. *Top right* the original signal $\mathbf{f}$ (*thin line*) versus the input signal $\mathbf{y}$ (*dashed line*) and the restored signal $\mathbf{s}_\rho$, optimal $\rho = 7.51$, (*thick line*). *Bottom left* DFT of $\mathbf{f}$ (*thin line*) versus DFT of $\mathbf{y}$ (*dashed line*) and of $\mathbf{s}_\rho$ (*thick line*). *Bottom right* fragments of those DFT

Hence, it follows that, if $\|\mathbf{y}\| > \varepsilon$ then Problem ORP has a unique solution $S_{\bar{\rho}}[\mathbf{y}, \mathbf{h}](t) \in {}^{2r}\mathscr{S}$, where the value of the parameter $\bar{\rho}$ is derived from the equation

$$\varepsilon(\rho) = \varepsilon^2. \tag{7.9}$$

The optimal regularization parameter was derived by the MATLAB function `defroP_conv.m`.

Figure 7.4 demonstrates the restoration of $\mathbf{f}$ by the spline $\mathbf{s}_\rho$ where $\rho = 7.51$ is derived from Eq. (7.9). We observe that noise is almost completely eliminated and the essential structure of $\mathbf{f}$ is resolved satisfactorily. The plots were produced by the MATLAB code `regExamplePopt.m`.

Approximation to the exact solution

The following fact was established in [1].

Theorem 7.1 *Assume that the functions f and h are T-periodic and belong to C^r and to C^l, respectively, and the Fourier coefficients $c_n(h) \neq 0$. Let $g(t) = h \circledast f(t)$. Then, $g \in C^{l+m}$. If the sampling step $1/N \to 0$ and the norm ε^2 of the error vector tends to zero then there exists a scheme for selection of the parameter $\rho(N, \varepsilon)$ such that*

$$\max_{i=0:r-1} \max_{t\in[0,T]} |S^{2r}_{\rho(N,\varepsilon)}(t)^{(i)} - f^{(i)}(t)| \to 0 \text{ as } N \to \infty,\ \varepsilon \to 0.$$

Noise estimation

We assume that the error vector $\mathbf{e}$ is a zero mean Gaussian white noise. Typically, the convolution kernel $h(t)$ is efficiently bandlimited. Then, its significant Fourier coefficients $c_n(h)$ occupy a relatively narrow band around zero, $-K < n < K$, $K < N/2$. Hence, the DFT of the data vector $\mathbf{y}$ $\left\{\hat{y}[n]\right\}_{n\in[K,N/2-1]\cup[-N/2,-K]} \approx \left\{\hat{e}[n]\right\}_{n\in[K,N/2-1]\cup[-N/2,-K]}$, where $\left\{\hat{e}[n]\right\}$ are the DFT of the error vector $\mathbf{e}$. By relying on the fact that the power spectrum $\left\{|\hat{e}[n]|^2\right\}$ of the white noise $\mathbf{e}$ is close to a constant for all $n = -N/2, ..., N/2-1$, we evaluate the variance

$$\mathrm{var}(\mathbf{e}) \approx \frac{1}{(N-K)^2} \sum_{n\in[K,N/2-1]\cup[-N/2,-K]} \left|\hat{y}[n]\right|^2, \Longrightarrow \varepsilon^2 = N\,\mathrm{var}(\mathbf{e}). \quad (7.10)$$

Another option for the noise estimation is to use the scheme in [3].

7.1.2 Solution of Two-Dimensional Convolution Equations

Similarly to the 1D case, we reduce Eq. (7.2) to the periodic convolution equation

$$g(x, y) = \int_0^T \int_0^T f(x-\xi, y-\eta)\, h(\xi, \eta)\, d\xi\, d\eta, \quad (7.11)$$

where the kernel h, the sought-for solution f and the output g are T-periodic with respect to both arguments x and y. For simplicity, we put $T = N = 2^j$. Assume $\mathbf{z} = \left\{z[k,n] = g(k,n) + e_{k,n}\right\}$, $k, n = 0, ..., N-1$, where $\mathbf{e} = \left\{e_{k,n}\right\}$, $k, n = 0, ..., N-1$, is the array of random errors, which we assume to be a zero-mean white noise. The arrays $\mathbf{z} = \{z[k,n]\}$ and $\mathbf{h} = \{h(k,n)\}$, $k, n = 0, ..., N-1$, are available data.

As in Sect. 5.2, $\zeta^p[\kappa, \nu](x, y)$ will stand for $\zeta^p[\kappa](x)\,\zeta^p[\nu](y)$ and $u^p[\kappa, \nu]$ will stand for $u^p[\kappa]\,u^p[\nu]$.

Let

$$\Upsilon(x, y) = \frac{1}{N} \sum_{\kappa,\nu=-N/2}^{N/2-1} \mu[\kappa, \nu]\, \zeta^q[\kappa, \nu](x, y) \in {}^{q,q}\mathscr{S}, \quad \mu[\kappa, \nu] = \frac{\hat{h}(\kappa, \nu)}{u^q[\kappa, \nu]},$$

be the spline from the space ${}^{q,q}\mathscr{S}$ that interpolates the sampled kernel $\mathbf{h}$.

7.1.2.1 Parameterized Spline

Like in the 1D case, the solution of the deconvolution problem is approximated by the spline $S_\rho(x, y) \in {}^{2r,2r}\mathscr{S}$, which minimizes the functional $J_\rho(S) \stackrel{\mathrm{def}}{=} \rho I(S) + E(S)$,

where

$$I(S) \stackrel{\text{def}}{=} \|S_x^{(r)}\|^2 + \|S_y^{(r)}\|^2, \quad E(S) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} (S \circledast \Upsilon(k,n) - z[k,n])^2 .$$

Let a spline $S(x, y) \in {}^{2r,2r}\mathscr{S}$ be represented as

$$S(x, y) = \frac{1}{N} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa, \nu] \;\; \zeta^{2r}[\kappa, \nu](x, y).$$

Then, Eq. (4.63) implies that

$$I(S) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \left(W^r[\kappa, \nu] + W^r[\nu, \kappa]\right) \; |\xi[\kappa, \nu]|^2,$$
$$W^r[\kappa, \nu] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi\kappa}{N}\right)^{2r} \; u^{2r}[\kappa]\, u^{4r}[\nu]. \tag{7.12}$$

Using Eq. (4.65), the sampled circular convolution is represented as

$$S \circledast \Upsilon(k, n) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \omega^{(k\kappa+n\nu)} \; \xi[\kappa, \nu]\, \mu[\kappa, \nu]\, u^{2r+q}[\kappa, \nu].$$

Hence, the discrepancy functional

$$E(S) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \left|\xi[\kappa, \nu]\, \mu[\kappa, \nu]\, u^{2r+q}[\kappa, \nu] - \hat{z}[\kappa, \nu]\right|^2 .$$

The parameterized functional to be minimized is

$$J_\rho(S) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \rho\, |\xi[\kappa, \nu]|^2 \left(W^r[\kappa, \nu] + W^r[\nu, \kappa]\right)$$
$$+ \left|\xi[\kappa, \nu]\, \hat{h}[\kappa, \nu]\, \frac{u^{2r+q}[\kappa, \nu]}{u^q[\kappa, \nu]} - \hat{z}[\kappa, \nu]\right|^2 .$$

Similarly to 1D case, the solution to the minimization problem is derived:

$$S_\rho[\mathbf{z}, \mathbf{h}](x, y) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \xi[\kappa, \nu](\rho)\, \zeta^p[\kappa, \nu](x, y),$$

$$\xi[\kappa,\nu](\rho)=\frac{\hat{z}[\kappa,\nu]\,\mu^*[\kappa,\nu]\,u^{2r+q}[\kappa,\nu]}{A_\rho[\kappa,\nu]},$$
$$A_\rho[\kappa,\nu]\stackrel{\text{def}}{=}\rho\left(W^r[\kappa,\nu]+W^r[\nu,\kappa]\right)+\left|\mu[\kappa,\nu]\,u^{2r+q}[\kappa,\nu]\right|^2.$$

The grid samples of the parameterized spline are calculated by using the 2D IDFT:

$$S_\rho[\mathbf{z},\mathbf{h}](k,n)=\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\xi[\kappa,\nu](\rho)\,u^{2r}[\kappa,\nu]\,\omega^{(\kappa k+\nu n)}$$
$$=\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\frac{\omega^{(\kappa k+\nu n)}\,\hat{z}[\kappa,\nu]\,\mu^*[\kappa,\nu]\,u^{2r}[\kappa,\nu]\,u^{2r+q}[\kappa,\nu]}{A_\rho[\kappa,\nu]}.$$

The spline values in between the grid points are calculated by the subdivision methods described in Chap. 6.

7.1.2.2 Optimal Regularization Parameter

Assume that the squared norm of the errors $\varepsilon^2\stackrel{\text{def}}{=}\sum_{k,n=0}^{N-1}e_{k,n}^2$ is estimated. Similarly to 1D case, the optimal regularization parameter ρ is derived from the equation

$$e(\rho)\stackrel{\text{def}}{=}E(S_\rho[\mathbf{z},\mathbf{h}])=\varepsilon^2=\sum_{k,n=0}^{N-1}e_{k,n}^2$$
$$\Longleftrightarrow\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\left(\frac{\rho\,\left|\hat{z}[\kappa,\nu]\right|\,\left(W^r[\kappa,\nu]+W^r[\nu,\kappa]\right)}{A_\rho[\kappa,\nu]}\right)^2=\varepsilon^2,\quad(7.13)$$

which has a unique solution.

The optimal regularization parameter was derived by the MATLAB function `defroP_conv.m`.

7.1.2.3 Examples

Performance of the 2D spline based Tikhonov deconvolution algorithm is illustrated on the images "Lena" and "Barbara", which are displayed in Fig. 7.5. The images are presented by arrays $\mathbf{f}=\{f[k,n]\}$ of pixels of size 512×512. The "Lena" image is more regular, while "Barbara" contains oscillating texture.

Prior to experiments, the images were blurred by convolving with the 2D Gaussian kernel

Fig. 7.5 Original images: On the *left* "Lena", on the *right* "Barbara"

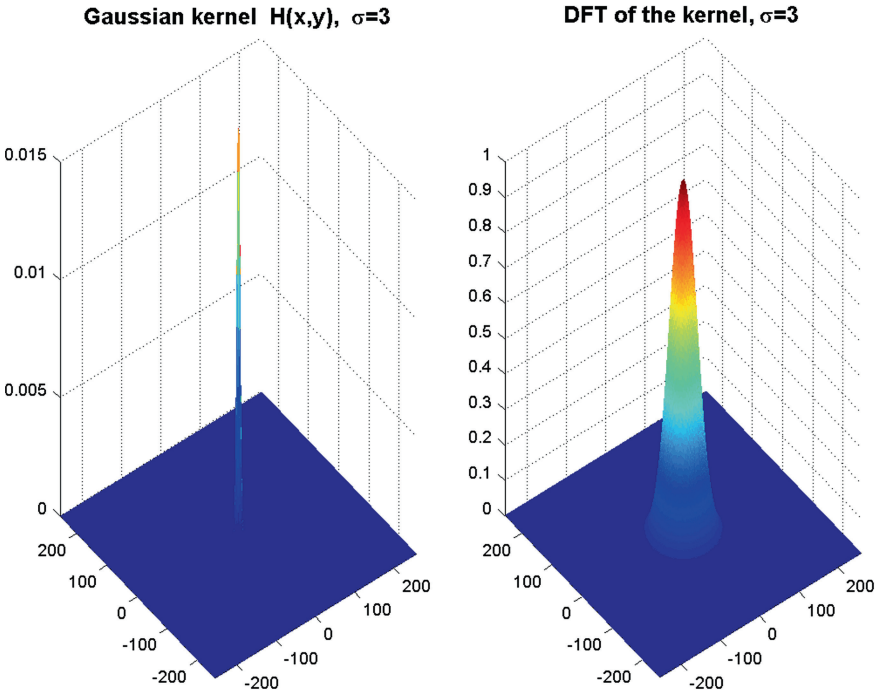


Fig. 7.6 On the *left* Gaussian kernel $h(x, y)$, $\xi = 3$, on the *right* its DFT

$$f(x, y) \longrightarrow g(x, y) = h \circledast f(x, y), \quad h(x, y) = e^{-\left(\frac{x^2}{2\sigma^2} + \frac{u^2}{2\sigma^2}\right)}, \ \sigma = 3,$$

(Matlab function `fspecial('gaussian')`). The sampled kernel and its 2D DFT are displayed in Fig. 7.6.

The blurred images are displayed in Fig. 7.7.

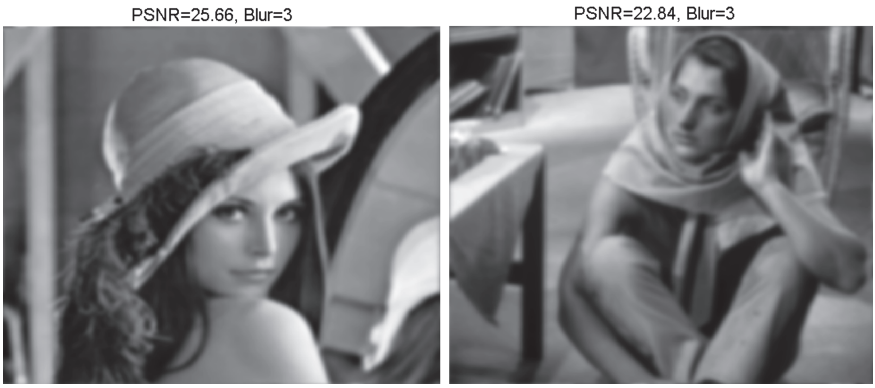


Fig. 7.7 Blurred images: On the *left* "Lena", PSNR = 25.66 dB on the *right*: "Barbara", PSNR = 22.84 dB

The following experiments on restoration of the images illustrate the performance of the algorithm:

Images restoration from noised data arrays

Pixels arrays $\mathbf{g} = \{g[k,n]\}$ from the blurred images were corrupted by zero-mean Gaussian noise $\mathbf{e} = \left\{e_{k,n}\right\}$ with different standard deviation (STD):

$$\mathbf{g} \longrightarrow \mathbf{z} = \left\{z[k,n] = g[k,n] + e_{k,n}\right\},\ k,n = 0,..,511.$$

Then, the deconvolution cubic splines $S_\rho[\mathbf{z},\mathbf{h}](x,y) \in {}^{4,4}\mathscr{S}$ were constructed, the parameter ρ was derived from Eq. (7.13). Thus, the images are approximated by the samples of the splines: $f[k,n] \approx S_\rho[\mathbf{z},\mathbf{h}](k,n)$.

Images restoration from downsampled kernel and noised data arrays

In these experiments, the sparse corrupted data and kernel arrays

$$\underline{\mathbf{z}} = \left\{\underline{z}[l,m] = g[rl,rn] + e_{l,m}\right\},\quad \underline{\mathbf{h}} = \left\{\underline{h}[l,m] = h[rl,rn]\right\},$$
$$l,m = 0,...,512/r-1,$$

are available. The images are approximated by the values of the splines: $f[k,n] \approx S_\rho[\underline{\mathbf{z}},\underline{\mathbf{h}}](k,n),\ k,n = 0,..,511$, which are constructed on the grid $\{rl,rn\},\ l,m = 0,...,512/r-1$. The splines values are calculated by using the dyadic subdivision procedures described in Chap. 6. In these experiments, the noise STD was fixed and, the sampling rate r varied.

Quality of restored images is evaluated visually and by the peak signal-to-noise ratio (PSNR) defined in Eq. (6.42).

Figures 7.8 and 7.9 illustrate restoration of the non-decimated "Lena" and "Barbara" images, respectively. The upper frames in each image demonstrate restoration of the images from the blurred ($\sigma = 3$) data, which was corrupted by a weak

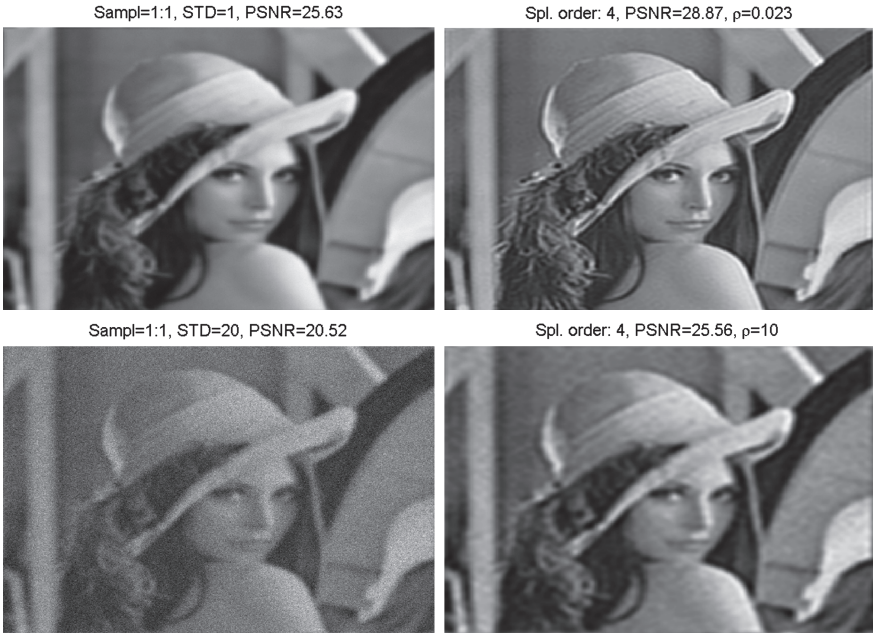


Fig. 7.8 *Top left* the blurred "Lena" image corrupted by noise whose STD is 1, PSNR = 25.63 dB. *Top right* the cubic deconvolution spline, PSNR = 28.87 dB. Bottom left: the image corrupted by noise, $\sigma = 20$, PSNR = 20.52 dB. Bottom right: the cubic deconvolution spline, PSNR = 25.56 dB

noise, whose STD is 1. In both cases, noise is eliminated and significant deblurring is achieved. The PSNR is elevated from 25.63 to 28.87 dB for "Lena" and from 22.82 to 23.86 dB for "Barbara". The bottom frames demonstrate restoration of the images from strongly noised (STD = 20) blurred data. PSNR of the corrupted "Lena" and "Barbara" images are 20.52 and 19.46 dB, respectively. The images restored by application of deconvolution splines have PSNR 25.56 dB for "Lena" and to 22.76 dB for "Barbara". The main structure of the images was restored, while noise was suppressed, which is achieved by using values of ρ that are much bigger than in the former case.

Figure 7.10 illustrates restoration of the "Lena" image when the decimated kernel and blurred noised data arrays are available. The downsampling rate is 1:2. As in the previous example, the upper frames in each image demonstrate restoration of the image from the decimated blurred ($\sigma = 3$) data, which was corrupted by a weak noise, whose STD is 1. The image is satisfactorily restored, and noise is completely eliminated. The PSNR is elevated from 25.64 to 28.33 dB. This PSNR number is close to the numbers, which were achieved in process of restoration of the images from non-decimated data in the previous example. The visual quality of the restored images in both examples is similar as well. The bottom frames illustrate the restoration when a stronger noise (STD is 20) corrupted the blurred decimated input data. In this

Fig. 7.9 *Top left* the blurred "Barbara" image corrupted by noise whose STD is 1, PSNR = 22.82 dB. *Top right* the cubic deconvolution spline, PSNR = 23.86 dB. *Bottom left* the image corrupted by noise, $\sigma = 20$, PSNR = 19.46 dB. *Bottom right* the cubic deconvolution spline, PSNR = 22.76 dB

case the restoration result is inferior compared to the result of the previous example. PSNR of the corrupted "Lena" decimated image is 20.54 dB. The images restored by application of deconvolution splines has PSNR of 24.47 dB.

Figures 7.8, 7.9 and 7.10 were produced by the MATLAB code `conv_exa2_stdP.m`. The forthcoming Figs. 7.11 and 7.12, which were produced by the MATLAB code `conv_exa2_sampP.m`, illustrate performance of the restoration algorithm on the blurred ($\sigma = 3$) decimated input data when noise is almost negligible. STD is being 0.01. The kernel and input data arrays are downsampled at the rate of 1:2 in the upper frames and 1:4 in the bottom frames. One can observe that, when sampling rate is 1:2, "Lena" is restored very close to the original, while fine texture in "Barbara" is somewhat blurred. The PSNR is elevated from 25.71 to 31.62 dB for "Lena" and from 22.79 to 24.42 for "Barbara". After decimation with the rate 1:4 "Lena" is restored still satisfactorily with the PSNR at 29.22 dB, while the texture in "Barbara" has degraded even more. The PSNR is elevated from 22.79 to 23.5 dB.

Comments

The above examples demonstrate that the parameterized deconvolution splines, where the regularization parameter ρ is derived automatically from Eq. (7.13) efficiently restores images from the non-decimated and decimated kernel and blurred noised data arrays when noise is not too strong. Otherwise, to suppress the noise,

Fig. 7.10 *Top left* the blurred "Lena" image downsampled at 1:2 and corrupted by noise whose STD is 1, PSNR = 25.64 dB. *Top right* the cubic deconvolution spline, PSNR = 28.33 dB. *Bottom left* the image corrupted by noise, $\varepsilon = 20$, PSNR = 20.54 dB. *Bottom right* cubic deconvolution spline, PSNR = 24.47 dB

the algorithm uses large values of the regularization parameter, which results in oversmoothing the images. Design of the 2D deconvolution spline is implemented by the MATLAB function `convsplaP2.m`.

A possible way to improve the performance is to use different values of ρ for different frequency domains of images. This approach, which is based on utilizing spline wavelet packets will be discussed in Sect. 11.1. Another way to improve the deconvolution results is application of the so-called Bregman Iteration technique [2, 8] using wavelet frames. That approach is implementing a l_1-minimization rather than the l_2-minimization, on which the Tikhonov regularization is based. Although the Bregman Iteration algorithm typically produces better deconvolution results compared to the Tikhonov method presented above, its computational cost is significantly higher.

7.2 Inversion of Heat Equation

Inverting the heat equation, which is an ill-conditioned problem, is closely related to solving the convolutionequation (7.1) (Eq. (7.11) in two dimensions).

Fig. 7.11 *Top left* the blurred "Lena" image downsampled at 1:2, noise STD is 0.01, PSNR = 25.66 dB. *Top right* the cubic deconvolution spline, PSNR = 31.24 dB. *Bottom left* the image downsampled at 1:4, PSNR = 25.5 dB. *Bottom right* the cubic deconvolution spline, PSNR = 27.26 dB

Assume that a T-periodic function $f(x)$, which defines an initial temperature distribution along a bar of length T, belongs to the space C^r, which is the space of T-periodic functions having continuous derivatives up to order r. Denote by $\mathbf{U_t}$ the linear operator such that $\mathbf{U}_t f(x) = g(x,t)$, where $g(x,t)$ is the T-periodic solution of the heat equation

$$\frac{\partial g(x,t)}{\partial t} = g''_x(x,t),\, g(x,0) = f(x). \tag{7.14}$$

The function $g(x,t)$ defines the temperature distribution along the bar at the moment $t > 0$. Figure 7.13 produced by the MATLAB code `heat_propP.m` illustrates the temperature distribution at different time moments.

In the two-dimensional setting, assume that a function $f(x,y)$, which defines an initial temperature distribution at the plate of size $T \times T$, is T-periodic in both directions and has continuous partial derivatives up to order r. Denote by $\mathbf{U}_t^2$ the linear operator such that $\mathbf{U}_t^2 f(x,u) = g(x,y,t)$ where $g(x,y,t)$ is the T-periodic solution of the heat equation

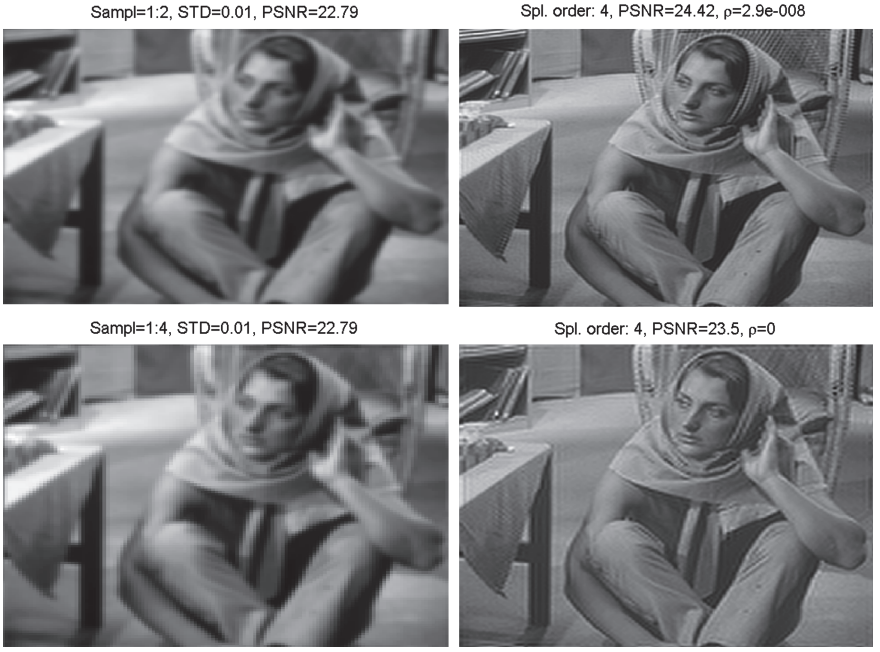


Fig. 7.12 *Top left* the blurred "Barbara" image downsampled at 1:2, noise STD is 0.01, PSNR = 22.79 dB. *Top right* the cubic deconvolution spline, PSNR = 24.42 dB. *Bottom left* the image downsampled at 1:4, PSNR = 22.79 dB. *Bottom right* the cubic deconvolution spline, PSNR = 23.5 dB

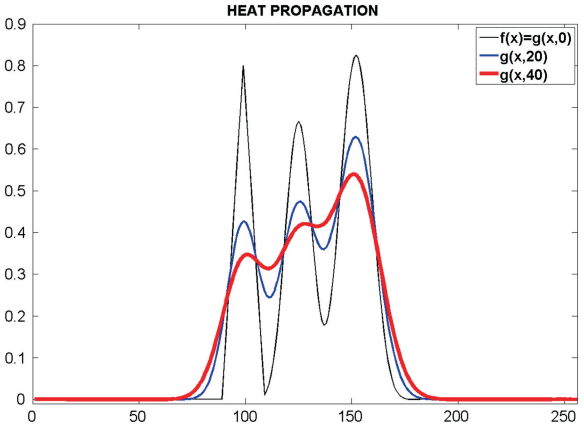


Fig. 7.13 *Thin line* initial temperature distribution $f(x) = g(x, 0)$. *Medium width line* temperature distribution $g(x, 20)$ at $t = 20$. Thick line: temperature distribution $g(x, 40)$ at $t = 40$

$$\frac{\partial g(x,y,t)}{\partial t} = g''_x(x,y,t) + g''_y(x,y,t), \quad g(x,y,0) = f(x,y). \tag{7.15}$$

The function $g(x, y, t)$ defines the temperature distribution at the plate at the moment $t > 0$.

Problem IP1: *Let t be a fixed time parameter. Given $g(x,t) = \mathbf{U}_t f(x)$,* find $f(x)$.
Problem IP2: *Let t be a fixed time parameter. Given $g(x,y,t) = \mathbf{U}_t^2 f(x,y)$,* find $f(x,y)$.

The problems have explicit theoretical solutions [4].

1D case

Assume for a moment that the T-periodic function $f(x)$ is known. Its Fourier expansion is

$$f(x) = \frac{1}{T}\sum_{n\in\mathbb{Z}} c_n(f)\, e^{2\pi i\, nx/T}.$$

The function $g(x,t)$ is T-periodic with respect to x and can be represented as

$$g(x,t) = \frac{1}{T}\sum_{n\in\mathbb{Z}} g_n(t)\, e^{2\pi i\, nx/T}, \quad g_n(0) = c_n(f). \tag{7.16}$$

Substituting the expression (7.16) into Eq. (7.14), we get

$$\frac{dg_n(t)}{dt} = -\left(\frac{2\pi\, n}{T}\right)^2 g_n(t), \quad g_n(0) = c_n(f) \Longrightarrow g_n(t) = c_n(f)\, e^{-t(2\pi\, n/T)^2}. \tag{7.17}$$

Now, assume that we know the function $g(x,t) = \mathbf{U}_t f(x)$ at some fixed time t, and Eq. (7.16) provides its Fourier expansion, where $g_n(t) = \int_0^T g(x,t)\, e^{-2\pi i\, nx/T}\, dx$. Then, Eq. (7.17) implies that

$$c_n(f) = g_n(t)\, e^{t(2\pi\, n/T)^2} \Longrightarrow f(x) = \frac{1}{T}\sum_{n\in\mathbb{Z}} g_n(t)\, e^{t(2\pi\, n/T)^2}\, e^{2\pi i\, nx/T}. \tag{7.18}$$

2D case

The Fourier expansion of the sought-for function $f(x, y)$ is:

$$f(x,y) = \frac{1}{T^2}\sum_{k,n\in\mathbb{Z}} c_{k,n}(f)\, e^{2\pi i(kx+ny)/T}.$$

The function $g(x, y, t)$ is T-periodic with respect to x and y and is represented as

$$g(x,y,t) = \frac{1}{T^2}\sum_{k,n\in\mathbb{Z}} g_{k,n}(t)\, e^{2\pi i(kx+ny)/T}, \quad g_{k,n}(0) = c_{k,n}(f). \tag{7.19}$$

Substituting the expression (7.19) into Eq. (7.15), we get

$$\begin{aligned} &\frac{dg_{k,n}(t)}{dt} = -\left(\frac{2\pi}{T}\right)^2 \left(k^2+n^2\right) g_{k,n}(t), \quad g_{k,n}(0) = c_{k,n}(f) \\ &\Longrightarrow g_{k,n}(t) = c_{k,n}(f)\, e^{-t(2\pi/T)^2\left(k^2+n^2\right)}. \end{aligned} \tag{7.20}$$

Assume that the function $g(x,y,t) = \mathbf{U}_t^2 f(x,y)$ at some fixed t is known and Eq. (7.19) provides its Fourier expansion, where $g_{k,n}(t) = \int_0^T \int_0^T g(x,y,t)\, e^{-2\pi i(kx+ny)/T}\, dx\, dy$. Then, Eq. (7.20) implies that

$$\begin{aligned} &c_{k,n}(f) = g_{k,n}(t)\, e^{t(2\pi/T)^2\left(k^2+n^2\right)} \\ &\Longrightarrow f(x,y) = \frac{1}{T^2}\sum_{k,n\in\mathbb{Z}} g_{k,n}(t)\, e^{t(2\pi/T)^2\left(k^2+n^2\right)}\, e^{-2\pi i(kx+ny)/T}. \end{aligned} \tag{7.21}$$

7.2.1 Inversion of One-Dimensional Heat Equation

In real life, the function $g(x,t)$ is known up to some errors: $y(x,t) = g(x,t) + e(x)$. Generally, there is no reason to assume that the Fourier coefficients $c_n(e)$ tend to zero faster than $e^{-n^2 t}$ (if they tend to zero at all). Therefore, application of direct inversion according to Eq. (7.18) to the available data $y(x,t)$

$$\begin{aligned} \tilde{f}(x) &= \frac{1}{T}\sum_{n\in\mathbb{Z}} y_n(t)\, e^{t(2\pi\, n/T)^2}\, e^{2\pi i\, nx/T} \\ &= \frac{1}{T}\sum_{n\in\mathbb{Z}} g_n(t)\, e^{t(2\pi\, n/T)^2}\, e^{2\pi i\, nx/T} + \frac{1}{T}\sum_{n\in\mathbb{Z}} c_n(e)\, e^{t(2\pi\, n/T)^2}\, e^{2\pi i\, nx/T} \end{aligned}$$

results in an unstable solution. However small the magnitude of the error function $|e(x)|$ is, the function $\tilde{f}(x)$ can comprise strong high-frequency components, which do not exist in the original function $f(x)$. Therefore, a regularization, which provides a stability to the solution at the expense of some deviation from the available data $y(x,t)$, is needed.

In most practical applications, the data function $g(x,t)$ is sampled on a grid $\{x_k\}$. Typically, the samples are corrupted by a random noise $\{e_k\}$. In this section, an approximate spline solution to Problem IP1 is derived. The solution is derived in a way similar to the solution of the deconvolution problem, where periodic splines are used.

7.2.1.1 Embedding the Problem into Spline Setting

Difference approximation

Denote by $\mathbf{V}_t^d$ the linear operator defined on the spline space ${}^{2r}\mathscr{S}$ such that for $S(x) \in {}^{2r}\mathscr{S}$, $\mathbf{V}_t^d S(x) = s(x,t)$, where $s(x,t)$ is the spline from ${}^{2r}\mathscr{S}$ (with respect to x), which provides a solution to the difference approximation of the heat equation

$$\frac{\partial s(x,t)}{\partial t} = \delta_x^2[s](x,t), \quad s(x,0) = S(x). \tag{7.22}$$

Here, $\delta_x^2[q](x) \stackrel{\text{def}}{=} (q(x-1) - 2q(x) + q(x+1)$ is the circular second central difference.

Assume the spline $S(x)$ is represented as

$$S(x) = \sum_{k=0}^{N-1} q[k]\, B^{2r}(x-k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta^{2r}[n](x), \quad \xi[n] = \hat{q}(n). \tag{7.23}$$

The splines $s(x,t) = \mathbf{V}_t S(x)$ can be represented in a similar form

$$s(x,t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi(t)[n]\, \zeta^{2r}[n](x), \quad \xi(0)[n] = \xi[n].$$

When the operator $\mathbf{V}_t^d$ is applied, it follows from Eqs. (4.12) and (7.22) that

$$\delta_x^2[s](x,t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi(t)[n]\, \delta_x^2[\zeta^{2r}[n]](x)$$

$$= -\frac{1}{N} \sum_{n=-N/2}^{N/2-1} w^2[n] \xi(t)[n]\, \zeta^{2r}[n](x), \quad w^2[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^2,$$

$$\Longrightarrow \frac{d\xi(t)[n]}{dt} = -w^2[n]\, \xi(t)[n], \quad \xi(0)[n] = \xi[n] \Longrightarrow \xi[n](t) = e^{-w^2[n]\, t} \xi[n].$$

Collocation approximation

Denote by $\mathbf{V}_t^c$ the linear operator on ${}^{2r}\mathscr{S}$ such that for $S(x) \in {}^{2r}\mathscr{S}$, $\mathbf{V}_t^c S(x) = s(x,t)$, where $s(x,t)$ is the spline from ${}^{2r}\mathscr{S}$, which satisfies the collocation conditions for the heat equation

$$\frac{\partial s(k,t)}{\partial t} = s_x''(k,t), \ k = 0, \ldots, N-1, \ s(x,0) = S(x). \tag{7.24}$$

Equations (4.8) and (4.11) imply that

$$\begin{aligned}\zeta^{2r}[n](k) &= \omega^{nk}u^{2r}[n], \quad \zeta^{2r}[n](x)^{(2)} = (1-\omega^{-n})^2\zeta^{2(r-1)}[n]\,(x+1)\\ &= (\omega^n - 2 + \omega^{-n})\zeta^{2(r-1)}[n](x) = -w^2[n]\zeta^{2(r-1)}[n](x),\\ \zeta^{2r}[n](k)^{(2)} &= -w^2[n]\,\zeta^{2(r-1)}[n](k) = -w^2[n]\,\omega^{nk}\,u^{2(r-1)}[n].\end{aligned}$$

Thus, Eq. (7.24) leads to the following relations for the coefficients

$$\begin{aligned}&\frac{d\xi(t)[n]}{dt} = -w^2[n]\frac{u^{2(r-1)}[n]}{u^{2r}[n]}\xi[n](t), \quad \xi(0)[n] = \xi[n]\\ &\Longrightarrow \xi[n](t) = \exp\left(-w^2[n]t\,\frac{u^{2(r-1)}[n]}{u^{2r}[n]}\right)\xi[n].\end{aligned}$$

Finally, application of the operators $\mathbf{V}_t = \mathbf{V}_t^{d(c)}$ to the spline $S \in {}^{2r}\mathscr{S}$ defined in Eq. (7.23) transforms its coefficients $\xi[n]$ into

$$\begin{aligned}&\xi[n](t) = \xi[n]\,\eta[n](t), \quad \eta[n](t) \stackrel{\text{def}}{=} e^{-b^r[n]\,t},\\ &b^r[n] \stackrel{\text{def}}{=} \begin{cases}(2\sin\pi n/N)^2\,, & \text{when}\mathbf{V}_t = \mathbf{V}_t^d\\ (2\sin\pi n/N)^2\,u^{2(r-1)}[n]/u^{2r}[n] & \text{when}\mathbf{V}_t = \mathbf{V}_t^c,\end{cases}\end{aligned} \tag{7.25}$$

Thus, similarly to the splines convolution, application of the operators $\mathbf{V}_t = \mathbf{V}_t^{d(c)}$ to a spline $S \in {}^{2r}\mathscr{S}$, results in multiplication of its coordinates $\xi[n]$ with some factors. Consequently, the scheme for finding approximate solutions is very similar to the scheme presented in Sect. 7.1. Note that, unlike convolution, the operators $\mathbf{V}_t$ map the space ${}^{2r}\mathscr{S}$ onto itself.

7.2.1.2 Parameterized Spline Solution

Assume that the sought-for initial temperature distribution function $f(x)$ is $T-$periodic and belongs to C^r. For simplicity, put $T = N$. Let $t > 0$ be a fixed time parameter. The data function $g(x,t) = \mathbf{U}_t f(x)$ is sampled on the uniform grid $\{k\},\ k = 0, ..., N-1,\ N = 2^j$, and the samples are available up to some errors. Thus, we have at our disposal a known time moment t and the data vector $\mathbf{y} = \{y[k] = g(k,t) + e_k\},\ k = 0, ..., N-1$. The function $f(x)$ is approximated by a spline $S_\rho(x) \in {}^{2r}\mathscr{S}$, which minimizes the functional $J_\rho(S) \stackrel{\text{def}}{=} \rho I(S) + E(S)$, where

$$I(S) \stackrel{\text{def}}{=} \|(S)^{(r)}\|^2, \quad E(S) \stackrel{\text{def}}{=} \sum_k (s(k,t) - y[k])^2\,, \quad s(x,t) \stackrel{\text{def}}{=} \mathbf{V}_t S(x)$$

and $\mathbf{V}_t = \mathbf{V}_t^{d(c)}$.

Let a spline $S(x) \in {}^{2r}\mathscr{S}$ be represented as

$$S(x) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi[n]\, \zeta^{2r}[n](x).$$

Then, similarly to Eq. (5.3), the functional $I(S)$ is

$$I(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} w^{2r}[n]\, u^{2r}[n]\, |\xi[n]|^2, \quad w^{2r}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N}\right)^{2r}.$$

It follows from Eq. (7.25) that the spline

$$\begin{aligned} s(x,t) &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \eta[n](t)\, \xi[n]\, \zeta^{2r}[n](x) \\ \Longrightarrow s(k,t) &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, u^{2r}[n]\, \eta[n](t)\, \xi[n]. \end{aligned}$$

Hence, using the Parseval identity (1.24), the discrepancy functional is expressed as

$$E(S) = \sum_{k=0}^{N-1} (s(k,t) - y[k])^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left| u^{2r}[n]\, \eta[n](t)\, \xi[n] - \hat{y}[n] \right|^2.$$

Consequently,

$$J_\rho(S) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \rho |\xi[n]|^2\, w^{2r}[n]\, u^{2r}[n] + \left| u^{2r}[n]\, \eta[n](t)\, \xi[n] - \hat{y}[n] \right|^2.$$

Similarly to Proposition 5.3, the following fact is derived.

Proposition 7.2 *Minimum of the functional $J_\rho(S)$ on the spline space ${}^{2r}\mathscr{S}$ is achieved by the spline*

$$\begin{aligned} &S_\rho[\mathbf{y}, t](x) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \xi_\rho[n]\, \zeta^{2r}[n](x), \quad \xi_\rho[n] = \frac{u^{2r}[n]\, \eta[n]^*(t)[n]\, \hat{y}[n]}{A_\rho[n]} \\ &A_\rho[n] \stackrel{\text{def}}{=} \rho\, u^{2r}[n]\, w^{2r}[n] + (|\eta[n](t)|\, u^{2r}[n])^2, \\ &S_\rho[\mathbf{y}, t](k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \frac{\omega^{kn} \left(u^{2r}[n]\right)^2\, \eta[n]^*(t)[n]\, \hat{y}[n]}{A_\rho[n]}. \end{aligned}$$

Design of the inversion D spline is implemented by the MATLAB function `convsplaP2.m`.

Optimal parameter

Similarly to the convolution equation, in a practical situation, when the data $\mathbf{y}$ is available and ε^2 is the estimate of the errors array squared norm, the parameter ρ can be selected as the solution to the equation

$$e(\rho) \stackrel{\text{def}}{=} E\left(S_\rho[\mathbf{y}, t]\right) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left(\frac{\rho\, u^{2r}[n]\, w^{2r}[n]\, |\hat{y}[n]|}{A_\rho[n])^2} \right)^2 = \varepsilon^2. \qquad (7.26)$$

The optimal regularization parameter is derived by the MATLAB function `defroP.m`.

Approximation to the exact solution

The splines $S_\rho[\mathbf{y}, t](x)$, where either of the operators $\mathbf{V}_t^d$ and $\mathbf{V}_t^g$ can be used, provide approximate solutions to Problem IP1 in the following sense:

Theorem 7.2 *Assume that the T-periodic function $f(x)$, which defines the initial condition in* (7.14)*, belongs to C^r. If the sampling step $1/N \to 0$ and the variance of the error vector* $\mathrm{var}(\mathbf{e}) \to 0$*, then there exists a scheme to select the parameter $\rho(N, e)$ such that*

$$\max_{i=0:r-1} \max_{x\in[0,1]} |S_{\rho(N,e)}[\mathbf{y}, t](x)^{(i)} - f^{(i)}(x)| \to 0 \text{ as } N \to \infty,\ e \to 0.$$

This fact was established in [9] for the case when the operator $\mathbf{V}_t^c$ was used. For the operator $\mathbf{V}_t^d$, the proof is similar.

7.2.2 Inversion of Two-Dimensional Heat Equation

Assume the time parameter t is fixed and samples $\{g(k, n, t)\}$, $k, n = 0, N-1$, of the function $g(x, y, t) = \mathbf{U}_t^2 f(x, y)$ are measured up to some zero-mean Gaussian errors $\mathbf{e} = \{e_{k,n}\}$. Thus, the array $\mathrm{z} = \{z[k, n] = g(k, n, t) + e_{k,n}\}$, $k, n = 0, N-1$, is available while the time moment t is known.

7.2.2.1 Embedding the Problem into Spine Setting

Difference approximation

Denote by $\mathbf{V}_t^{2d}$ the linear operator defined on the spline space ${}^{2r,2r}\mathscr{S}$ such that for $S(x, y) \in {}^{2r,2r}\mathscr{S}$, $\mathbf{V}_t^{2d} S(x, y) = s(x, y, t)$, where $s(x, y, t)$ is the spline from ${}^{2r,2r}\mathscr{S}$ (with respect to x, y), which provides a solution to the difference approximation of the heat equation

$$\frac{\partial s(x,y,t)}{\partial t} = \delta_x^2[s](x,y,t) + \delta_y^2[s](x,y,t), \quad s(x,y,0) = S(x,y). \tag{7.27}$$

Here, $\delta_x^2[q](x,y) \stackrel{\text{def}}{=} (q(x-1,y) - 2q(x,y) + q(x+1,y)$ and $\delta_y^2[q](x,y) \stackrel{\text{def}}{=} (q(x,y-1) - 2q(x,y) + q(x,y+1)$ are the partial circular second central differences.

Collocation approximation

Denote by $\mathbf{V}_t^{2c}$ the linear operator on ${}^{2r,2r}\mathscr{S}$ such that for $S(x,y) \in {}^{2r,2r}\mathscr{S}$, $\mathbf{V}_t^{2c} S(x,y) = s(x,y,t)$, where $s(x,y,t)$ is the spline from ${}^{2r,2r}\mathscr{S}$, which satisfies the collocation conditions for the heat equation

$$\frac{\partial s(k,n,t)}{\partial t} = s_x''(k,n,t) + s_y''(k,n,t), \ k = 0,\dots,N-1, \ s(x,y,0) = S(x,y). \tag{7.28}$$

Actually, application of the operators $\mathbf{V}_t^2 = \mathbf{V}_t^{2d(2c)}$ to the spline $S \in {}^{2r,2r}\mathscr{S}$ is similar to convolving it with another spline. Let the spline $S \in {}^{2r,2r}\mathscr{S}$ be represented as

$$S(x,y) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa,\nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y). \tag{7.29}$$

The splines $s(x,y,t) = \mathbf{V}_t^2 S(x,y)$ can be represented in a similar form

$$s(x,y,t) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi(t)[\kappa,\nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y), \quad \xi(0)[\kappa,\nu] = \xi[\kappa,\nu].$$

When the operator $\mathbf{V}_t^{2d}$ is applied, it follows from Eqs. (4.12) and (7.27) that

$$\begin{aligned}
\delta_x^2[s](x,y,t) &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi(t)[\kappa,\nu]\, \delta_x^2[\zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y) \\
&= -\frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} w^2[\kappa]\, \xi(t)[\kappa,\nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y), \\
\delta_y^2[s](x,y,t) &= -\frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} w^2[\nu]\, \xi(t)[\kappa,\nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y).
\end{aligned}$$

Hence, it follows that

$$\begin{aligned}
&\frac{d\xi(t)[\kappa,\nu]}{dt} = -\left(w^2[\kappa] + w^2[\nu]\right) \xi[\kappa,\nu](t), \quad \xi(0)[\kappa,\nu] = \xi[\kappa,\nu] \\
&\Longrightarrow \xi[\kappa,\nu](t) = e^{-t\left(w^2[\kappa]+w^2[\nu]\right)} \xi[\kappa,\nu], \quad w[\nu] \stackrel{\text{def}}{=} 2\sin\frac{\pi\nu}{N}.
\end{aligned}$$

Similarly to the 1D case, we get for the operator $\mathbf{V}_t^{2c}$

$$\begin{aligned}
&\zeta^{2r}[\kappa](k)\,\zeta^{2r}[\nu](n) = \omega^{(\kappa k+\nu n)}\,u^{2r}[\kappa]\,u^{2r}[\nu],\\
&\zeta^{2r}[\kappa](k)^{(2)}\,\zeta^{2r}[\nu](n) = -w^2[\kappa]\omega^{(\kappa k+\nu n)}\,u^{2(r-1)}[\kappa]\,u^{2r}[\nu],\\
&\zeta^{2r}[\kappa](k)\,\zeta^{2r}[\nu](n)^{(2)} = -w^2[\nu]\,\omega^{(\kappa k+\nu n)}\,u^{2(r-1)}[\nu]\,u^{2r}[\kappa].
\end{aligned}$$

Equation (7.28) leads to the following relations for the coefficients

$$\begin{aligned}
&\frac{d\xi(t)[\kappa,\nu]}{dt} = -\left(w^2[\kappa]\frac{u^{2(r-1)}[\kappa]}{u^{2r}[\kappa]} + w^2[\nu]\frac{u^{2(r-1)}[\nu]}{u^{2r}[\nu]}\right)\xi[\kappa,\nu](t),\\
&\xi(0)[\kappa,\nu] = \xi[\kappa,\nu]\\
&\Longrightarrow \xi[\kappa,\nu](t) = \exp\left(-t\left(w^2[\kappa]\frac{u^{2(r-1)}[\kappa]}{u^{2r}[\kappa]} + w^2[\nu]\frac{u^{2(r-1)}[\nu]}{u^{2r}[\nu]}\right)\right).
\end{aligned}$$

Finally, application of the operators $\mathbf{V}_t^2 = \mathbf{V}_t^{2d(2c)}$ to the spline $S \in {}^{2r}\mathscr{S}$ defined in Eq. (7.29) transforms its coefficients $\xi[\kappa,\nu]$ into

$$\begin{aligned}
&\xi[\kappa,\nu](t) = \xi[\kappa,\nu]\,\eta[\kappa](t)\,\eta[\nu](t), \quad \eta[n](t) \stackrel{\text{def}}{=} e^{-b^r[n]\,t},\\
&b^r[n] \stackrel{\text{def}}{=} \begin{cases} (2\sin\pi n/N)^2, & \text{when}\mathbf{V}_t^2 = \mathbf{V}_t^{2d}\\ (2\sin\pi n/N)^2\,u^{2(r-1)}[n]/u^{2r}[n] & \text{when}\mathbf{V}_t^2 = \mathbf{V}_t^{2c}, \end{cases}
\end{aligned} \tag{7.30}$$

Thus, application of the operators $\mathbf{V}_t^2$ to a spline $S \in {}^{2r,2r}\mathscr{S}$, like 2D convolution, results in multiplication of its coordinates $\xi[\kappa,\nu]$ with some factors. Consequently, the scheme for finding approximate solutions is very similar to the scheme presented in Sect. 7.1.2.

7.2.2.2 Parameterized Spline Solution

Let $t > 0$ be a fixed time parameter. The data function $g(x,y,t) = \mathbf{U}_t^2 f(x,y)$ is sampled on the uniform grid $\{k,n\}$, $k,n = 0,...,N-1$, $N = 2^j$ and the samples are available up to some errors. Thus, a known time moment t and the data array $\mathbf{z} = \left\{z[k,n] = g(k,n,t) + e_{k,n}\right\}$, $k,n = 0,...,N-1$, are available, where $\mathbf{e} = \left\{e_{k,n}\right\}$, $k,n = 0,...,N-1$, is the array of random errors, which is assumed to be a zero-mean white noise. The function $f(x,y)$ is approximated by a spline $S_\rho \in {}^{2r,2r}\mathscr{S}$, which minimizes the functional $J_\rho(S) \stackrel{\text{def}}{=} \rho I(S) + E(S)$, where

$$I(S) \stackrel{\text{def}}{=} \|S_x^{(r)}\|^2 + \|S_y^{(r)}\|^2, \quad E(S) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1}\left(s(k,n,t) - z[k,n]\right)^2,$$

where $s(x, y, t) \stackrel{\text{def}}{=} \mathbf{V}_t^2 S(x, y)$ and $\mathbf{V}_t^2 = \mathbf{V}_t^{2d(2c)}$.

Let the splines $S(x, y) \in {}^{2r,2r}\mathscr{S}$ and $s(x, y, t)$ be represented as

$$S(x, y) = \frac{1}{N} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi[\kappa, \nu] \;\; \zeta^{2r,2r}[\kappa, \nu](x, y),$$

$$\begin{aligned} s(x, y, t) &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \xi(t)[\kappa, \nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y) \\ &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \eta[\kappa](t)\, \eta[\nu](t)\, \xi[\kappa, \nu]\, \zeta^{2r}[\kappa](x)\, \zeta^{2r}[\nu](y). \end{aligned}$$

Then, as in Eq. (7.12)

$$I(S) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \left(W^r[\kappa, \nu] + W^r[\nu, \kappa]\right) |\xi[\kappa, \nu]|^2,$$

$$W^r[\kappa, \nu] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi\kappa}{N}\right)^{2r} u^{2r}[\kappa]\, u^{4r}[\nu].$$

It follows from Eq. (4.61) that the sampled spline

$$s(k, n, t) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \eta[\kappa](t)\, \eta[\nu](t)\, \xi[\kappa, \nu]\, u^{2r}[\kappa]\, u^{2r}[\nu]\, \omega^{(\kappa k+\nu n)}.$$

Hence, the discrepancy functional

$$E(S) = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \left|\eta[\kappa](t)\, \eta[\nu](t)\, \xi[\kappa, \nu]\, u^{2r}[\kappa]\, u^{2r}[\nu] - \hat{z}[\kappa, \nu]\right|^2.$$

Consequently,

$$\begin{aligned} J_\rho(S) = {} & \tfrac{1}{N^2} \textstyle\sum_{\kappa,\nu=-N/2}^{N/2-1} \rho\, |\xi[\kappa, \nu]|^2 \left(W^r[\kappa, \nu] + W^r[\nu, \kappa]\right) \\ & + \left|\eta[\kappa](t)\, \eta[\nu](t)\, \xi[\kappa, \nu]\, u^{2r}[\kappa]\, u^{2r}[\nu] - \hat{z}[\kappa, \nu]\right|^2. \end{aligned}$$

Similarly to the 1D case, the solution to the minimization problem is derived as

$$S_\rho[\mathbf{z}, t](x, y) = \frac{1}{N^2} \sum_{\kappa,\nu=0}^{N-1} \xi[\kappa, \nu](\rho)\, \zeta^p[\kappa, \nu](x, y),$$

$$\xi[\kappa,\nu](\rho)=\frac{\hat{z}[\kappa,\nu]\eta[\kappa]^*(t)\,\eta[\nu]^*(t)\,u^{2r}[\kappa]\,u^{2r}[\nu]}{A_\rho[\kappa,\nu]},$$

$$A_\rho[\kappa,\nu]\stackrel{\text{def}}{=}\rho\left(W^r[\kappa,\nu]+W^r[\nu,\kappa]\right)+\left|\eta[\kappa](t)\,\eta[\nu](t)\,u^{2r}[\kappa]\,u^{2r}[\nu]\right|^2.$$

Grid samples of the spline are

$$S_\rho[\mathbf{z},t](k,n)=\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\xi[\kappa,\nu](\rho)\,u^{2r}[\kappa]\,u^{2r}[\nu]\,\omega^{(\kappa k+\nu n)}$$

$$=\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\frac{\omega^{(\kappa k+\nu n)}\,\hat{z}[\kappa,\nu]\,\eta[\kappa]^*(t)\,\eta[\nu]^*(t)\,(u^{2r}[\kappa]\,u^{2r}[\nu])^2}{A_\rho[\kappa,\nu]}.$$

Optimal regularization parameter

Assume that the squared norm of the errors $\varepsilon^2\stackrel{\text{def}}{=}\sum_{k,n=0}^{N-1}e_{k,n}^2$ is estimated. Similarly to 1D case, the optimal regularization parameter ρ is derived from the equation

$$e(\rho)\stackrel{\text{def}}{=}E(S_\rho[\mathbf{z},t])=\varepsilon^2=\sum_{k,n=0}^{N-1}\varepsilon_{k,n}^2$$

$$\Longleftrightarrow\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\left|\hat{z}[\kappa,\nu]\right|^2\left(\frac{\rho\left(W^r[\kappa,\nu]+W^r[\nu,\kappa]\right)}{A_\rho[\kappa,\nu]}\right)^2=\varepsilon^2,\quad(7.31)$$

which has a unique solution.

The optimal regularization parameter is derived by the MATLAB function `defroP.m`.

7.2.2.3 Examples

Performance of the spline-based algorithm for inversion of the 2D heat equation is illustrated on the examples where the initial temperature distributions $f(x,y)$ are defined by the images "Lena" and "Barbara". The images are displayed in Fig. 7.5. The images are presented by arrays $\mathbf{f}=\{f[k,n]\}$ of pixels of size 512×512. Prior to experiments, the operator $\mathbf{U}_t^2$ defined in Eq. (7.15), with $t=5$, was applied to the original data. This operation resulted in blurring the images. The blurred images $g(x,y,t)=\mathbf{U}_t^2 f(x,y)$ are displayed in Fig. 7.14.

The following experiments illustrate the performance of the presented inversion algorithms:

Restoration of the initial distribution from noised data arrays

Pixels arrays $\mathbf{g}=\{g[k,n,t]\}$ from the blurred images were corrupted by zero-mean Gaussian noise $\mathbf{e}=\left\{e_{k,n}\right\}$ with different STD:

Fig. 7.14 Blurred images: On the *left* "Lena", PSNR = 25.65 dB, on the *right* "Barbara", PSNR = 22.84 dB

$$\mathbf{g} \longrightarrow \mathbf{z} = \left\{z[k,n] = g[k,n,t] + e_{k,n}\right\},\ k,n = 0,...511.$$

Then, the cubic splines $S_\rho[\mathbf{z},t](x,y) \in {}^{4,4}\mathscr{S}$ were constructed using the collocation and the difference approximations, where the parameter ρ values were derived from Eq. (7.31). The images were approximated by the samples of the splines: $f[k,n] \approx S_\rho[\mathbf{z},t](k,n)$.

Restoration of the initial distribution from downsampled data arrays

In these experiments, the sparse corrupted data arrays affected by weak noise

$$\underline{\mathbf{z}} = \left\{\underline{z}[l,m] = g[rl,rm,t] + e_{l,m}\right\},\ l,m = 0,...,512/r-1,$$

are available. The images were approximated by the values of the splines: $f[k,n] \approx S_\rho[\underline{\mathbf{z}},t](k,n),\ k,n = 0,...511$, which are constructed on the sparse grid $\{rl,rm\}$. The splines values are calculated by using the dyadic subdivision procedures described in Chap. 6.
Quality of restored distributions is evaluated visually and by the peak signal to noise ratio (PSNR) defined in Eq. (6.42).

RESTORATION RESULTS

Restoration from noised data arrays

Figures 7.15 and 7.16 illustrate the restoration of the non-decimated "Lena" and "Barbara" images, respectively. The left-hand frames in each image demonstrate restoration of the images from the blurred ($t = 5$) data, which was corrupted by a moderate noise, whose STD is 1. In both cases, noise is eliminated and significant deblurring is achieved. The PSNR is elevated from 25.63 to 28.78 dB for "Lena" and from 22.83 to 23.84 dB for "Barbara" using the collocation approximation. Utilizing the difference approximation produced very close results. The right-hand frames

Fig. 7.15 *Top left* the blurred "Lena" image ($t = 5$) corrupted by noise whose STD is 1, PSNR = 25.63 dB. *Top right* the blurred "Lena" image ($t = 5$) corrupted by noise whose STD is 20, PSNR = 20.53 dB. *Center left* the image restored by the cubic spline $S_\rho[\mathbf{z}, t](k, n)$, PSNR = 28.78 dB (collocation approximation). *Bottom left* the same with the difference approximation, PSNR = 28.7 dB. *Center right* the image restored by the cubic spline $S_\rho[\mathbf{z}, t](k, n)$, PSNR = 25.45 dB (collocation approximation). Bottom right: the same with the difference approximation, PSNR = 25.45 dB

demonstrate restoration of the images from strongly noised (STD is 20 for "Lena" and STD is 10 for "Barbara") blurred data. PSNRs of the corrupted "Lena" and "Barbara" images are 20.53 and 21.71 dB, respectively. The restored images have PSNR of 25.45 dB for "Lena" and 23.11 dB for "Barbara". The main structure of the images was restored. The collocation and the difference approximation produced the same PSNR results.

The experiments were implemented by the MATLAB code `heat_exa2_di_co_stdP.m`.

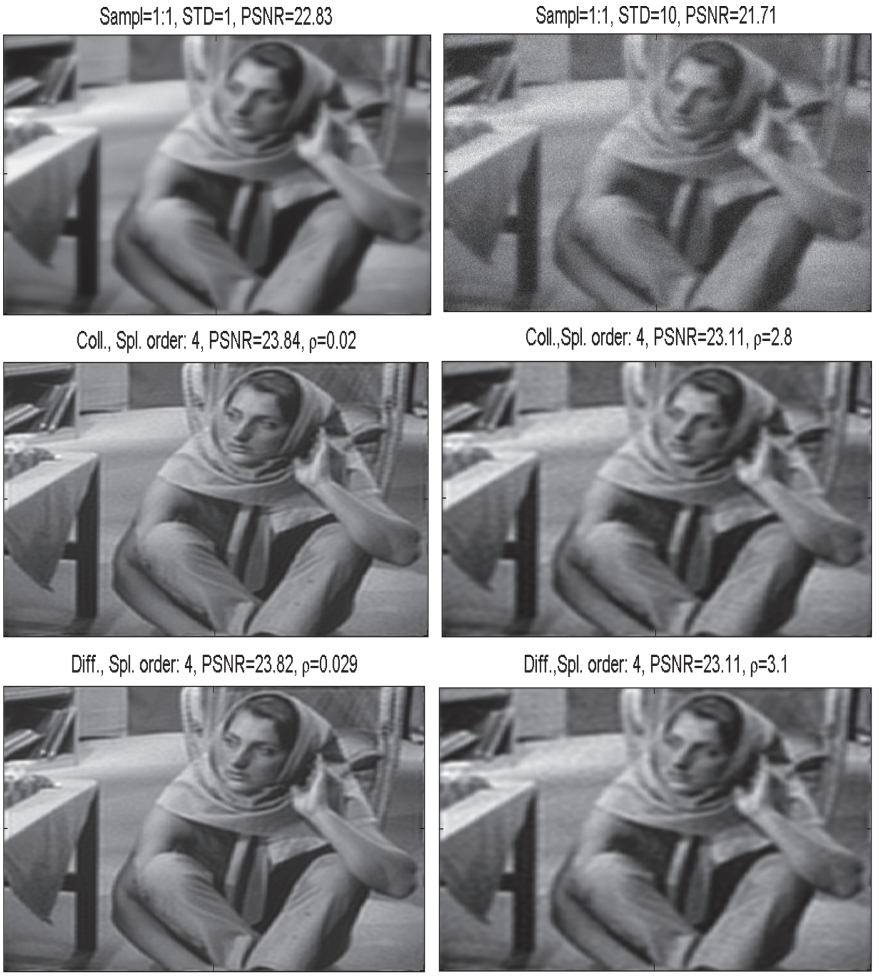


Fig. 7.16 *Top left* the blurred "Barbara" image ($t = 5$) corrupted by noise whose STD is 1, PSNR = 22.83 dB. *Top right* the blurred "Barbara" image ($t = 5$) corrupted by noise whose STD is 10, PSNR = 21.71 dB. *Center left* the image restored by the cubic spline $S_\rho[\mathbf{z}, t](k, n)$, PSNR = 23.84 dB (collocation approximation). *Bottom left* the same with the difference approximation, PSNR = 23.82 dB. *Center right* the image restored by the cubic spline $S_\rho[\mathbf{z}, t](k, n)$, PSNR = 23.11 dB (collocation approximation). *Bottom right* the same with the difference approximation, PSNR = 23.11 dB

Restoration from downsampled data arrays

In the examples, which are illustrated in Figs. 7.17 and 7.18, the performance of the inversion splines $S_\rho[\mathbf{z}, t](k, n)$, which are constructed using the *collocation* procedure is compared with the performance of the splines constructed using the *difference* procedure. It is done on the blurred ($t = 5$) decimated input data when noise is almost negligible: STD is 0.01. The input data arrays are downsampled at the rate of 1:2

Fig. 7.17 *Top left* the blurred "Lena" image downsampled at 1:2, noise STD is 0.01, PSNR = 25.68 dB. *Top right* the image downsampled at 1:4, PSNR = 25.59 dB. *Center* the eight-order inversion splines designed by the *collocation*. *Left* sampling 1:2, PSNR = 31.47 dB. *Right* sampling 1:4, PSNR = 29.3 dB. *Bottom* eight-order inversion splines constructed by the *difference*. *Left*–sampling 1:2, PSNR = 31.51 dB; *Right* sampling 1:4, PSNR = 29.21 dB

in the left frames and 1:4 in the right frames. In both cases, the eight-order splines constructed on the sparse grids and upsampled by the dyadic subdivision algorithm demonstrated the best results. When sampling rate was 1:2, the *difference* splines produced a little bit higher PSNR than the *collocation* splines and vice versa for the sampling rate of 1:4. For the "Lena" image the achieved PSNR values are 31.51 dB

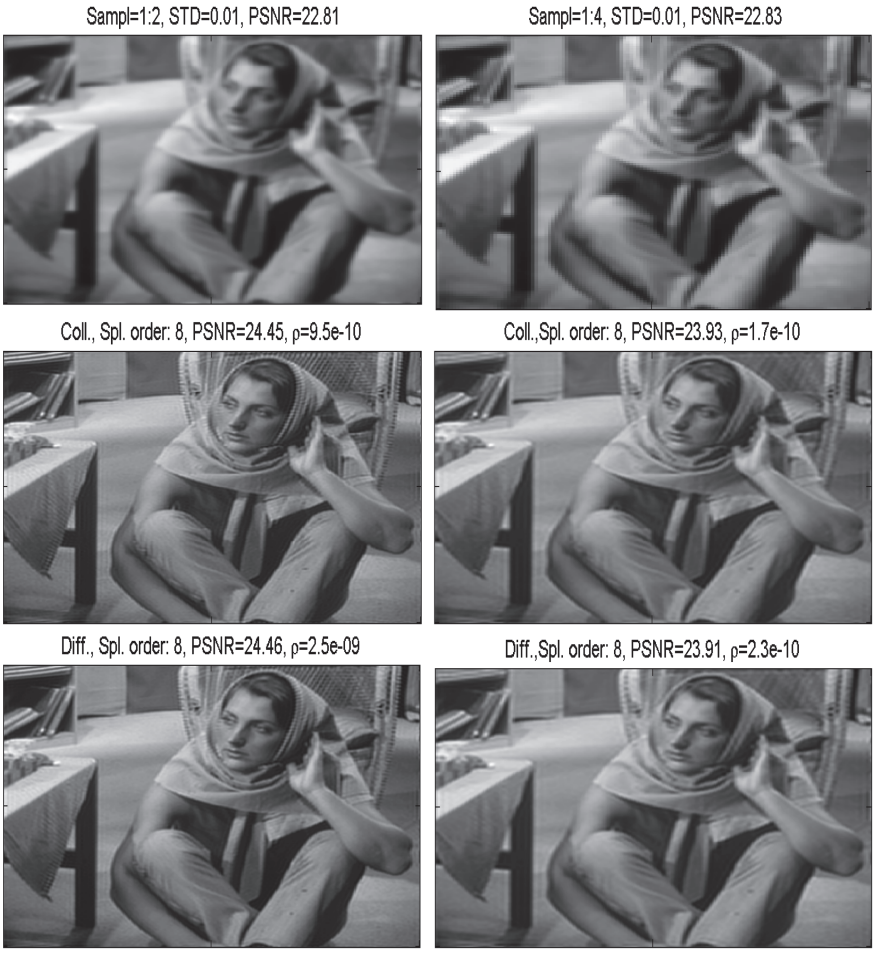


Fig. 7.18 *Top left* the blurred "Barbara" image downsampled at 1:2, noise STD is 0.01, PSNR = 22.81 dB. *Top right* the image downsampled at 1:4, PSNR = 22.83 dB. *Center* eight-order inversion splines constructed by the *collocation*. *Left* sampling 1:2, PSNR = 24.45 dB. *Right* sampling 1:4, PSNR = 23.93 dB. *Bottom* eight-order inversion splines constructed by the *difference*. *Left* sampling 1:2, PSNR = 24.46 dB. *Right* sampling 1:4, PSNR = 23.91 dB

(1:2) and 29.3 dB (1:4). For the "Barbara" image the achieved PSNR values are 24.46 dB (1:2) and 23.93 dB (1:4).

The experiments were implemented by the MATLAB code `heat_exa2_di_coP.m`.

Comments

The above examples demonstrate that the parameterized inversion splines, where the regularization parameter ρ is derived automatically from Eq. (7.31), efficiently

restores images from the non-decimated and decimated blurred noised data arrays when noise is not too strong. Otherwise, to suppress the noise, the algorithm uses large values of the regularization parameter, which results in oversmoothing the images. Design of the inversion 2D spline is implemented by the MATLAB function `heatsplaP2.m`.

A possible way to improve the performance is to use different values of ρ for different frequency domains of images. This approach, which is based on utilizing spline wavelet packets will be discussed in Sect. 11.2.

References

1. A. Averbuch, V. Zheludev, Spline-based deconvolution. Signal Process. **89**(9), 1782–1797 (2009)
2. J. Cai, S. Osher, Z. Shen, Split Bregman methods and frame based image restoration. Multiscale Model. Simul. **8**(2):337–369 (2009/10)
3. D. Donoho, I. Johnstone, Ideal spatial adaptation via wavelet shrinkage. Biometrika **81**(3), 425–455 (1994) NULL
4. J. Fourier. Théorie analytique de la chaleur. in *Firmin Didot*, (1822). (Cambridge University Press, Paris, 2009) (Reissued)
5. A.N. Tikhonov, Solution of incorrectly formulated problems and the regularization method. Soviet Math. Dokl. **4**(4), 1035–1038 (1963)
6. A.N. Tikhonov, V.Y. Arsenin, *Solutions of ill-posed problems* (Wiley, New York, 1977)
7. N. Wiener, *Extrapolation, interpolation, and smoothing of stationary time series with engineering applications* (Wiley, New York, 1949)
8. W. Yin, S. Osher, D. Goldfarb, J. Darbon, Bregman iterative algorithms for l_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**(1), 143–168 (2008)
9. V. A. Zheludev, in *Proceedings of the International Conference Approximation Theory* (1990), *Colloquium Mathematicum Society of J Bolyai*, ed. by J. Szabados, K. Tandoi (eds.). Spline-Operational Calculus and Inverse Problem for Heat Equation, vol. 58 (Kecskemét, Hungary, 1991), pp. 763–783

Chapter 8
Periodic Spline Wavelets and Wavelet Packets

Abstract This chapter presents wavelets and wavelet packets in the spaces of periodic splines of arbitrary order, which, in essence, are the multiple generators for these spaces. The SHA technique provides explicit representation of the wavelets and wavelet packets and fast implementation of the transforms in one and several dimensions.

A breakthrough in the spline theory and applications has taken place during recent decades, with the introduction of multiple generators of spline spaces, the so-called spline-wavelets and wavelet packets [2, 3, 6, 12, 13]. The components of these multiple generators are orthogonal to each other. In a sense, these multiple generators provide a representation of a spline, which is intermediate between a time- domain and a frequency domain representations. Once the spline is expanded over the shifts of the components of such a multiple generator, its Fourier spectrum becomes split into bands while the expansion coefficients retain a spatial meaning. This fact is advantageous for signal and image processing. The wavelets and wavelet packets achieve a remarkable success in denoising, data compression, convolution-related problems, and features extraction, to name a few.

Throughout this section it is assumed that the splines orders are even: $p = 2r$. Periodic setting provides much more flexibility compared to non-periodic setting in the design and implementation of spline wavelet and wavelet packet transforms [7, 8, 15, 16]. It stems from the fact that all the calculations consist of fast direct and inverse Fourier transforms (FFT and IFFT, respectively) and simple arithmetic operations. In particular, operations with the shift-orthogonal splines and the (Battle-Lemarié) wavelets become practical. Raising the splines order does not increase computation complexity.

Design of multiple generators for spline spaces starts with introduction of the splines spaces of different resolution.

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_8,

8.1 Spline Spaces of Different Resolution Scales

In Chap. 6, a sequence of the spline spaces, whose resolution was higher compared to the original space, was introduced. In this section, we move in the opposing direction by extending the sequence into lower resolution spaces.

Logically, we should use negative integers for indexing the lower resolution spaces. However, in order to avoid cumbersome notations, we use in this and the subsequent chapters positive indices (while hoping that this will not lead to confusion).

As before, $N = 2^j,\ \ j \in \mathbb{N}$ and $\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$. Denote $N_m \stackrel{\text{def}}{=} N/2^m = 2^{j-m}z$, $m = 0, 1, ...$

Denote by the symbol ${}^p\mathcal{S}_m$ the space of $N-$periodic splines of order p whose nodes are located on the grid $\{2^m k\},\ k \in \mathbb{Z},\ m = 0, 1, \ldots.$ The dimension of the space ${}^p\mathcal{S}_m$ is N_m. For the space ${}^p\mathcal{S}_0$ of $N-$periodic splines with nodes on the grid $\{k\},\ k \in \mathbb{Z}$, the notation ${}^p\mathcal{S}$ is retained. Obviously, for $m > 0$,

$$ {}^p\mathcal{S}_m \subset {}^p\mathcal{S}_{m-1} \ldots \subset {}^p\mathcal{S}_0 = {}^p\mathcal{S}. \tag{8.1}$$

The N-periodic normalized B-spline of the first order on the grid $\{2^m k\}$ and its Fourier coefficients are

$$B_m^1(t) \stackrel{\text{def}}{=} \sum_{k\in\mathbb{Z}} 2^{-m}\chi[-2^{m-1}, 2^{m-1}]\,(t + kN)\,, \quad c_n(B_m^1) = \frac{\sin \pi n/N_m}{\pi n/N_m},$$

where the characteristic function $\chi[a, b](t)$ is defined in Eq. (3.12). The N-periodic normalized B-spline of order p is defined via iterated circular convolution

$$B_m^p(t) \stackrel{\text{def}}{=} B_m^1 \circledast B_m^{p-1}(t) = \frac{1}{N}\sum_{n\in\mathbb{Z}} \left(\frac{\sin(\pi n/N_m)}{\pi n/N_m}\right)^p e^{2\pi i n t/N}. \tag{8.2}$$

The B-spline $B_m^p(t)$ belongs to the space ${}^p\mathcal{S}_m$, and its shifts form a basis of this space. Each spline $S(t) \in {}^p\mathcal{S}_m$ is represented as

$$S(t) = \sum_{k=0}^{N_m-1} q[k]B_m^p\left(t - 2^m k\right). \tag{8.3}$$

Similarly to the space ${}^p\mathcal{S}$, we introduce the orthogonal bases of exponential splines in ${}^p\mathcal{S}_m$:

$$\zeta_m^p[n](t) \stackrel{\text{def}}{=} \sum_{k=0}^{N_m-1} \omega^{2^m nk} B_m^p(t - 2^m k)$$

$$= \frac{1}{2^m} \sum_{l=-\infty}^{\infty} e^{2\pi i (n/N + 2^{-m} l)t} \left(\frac{\sin \pi (n/N_m + l)}{\pi (n/N_m + l)} \right)^p, \tag{8.4}$$
$$n = -N_m/2, \ldots, N_m/2 - 1.$$

The sequence $\{\zeta_m^p[n](t)\}$ of mutually orthogonal splines is N_m-periodic: $\zeta_m^p[n](t) = \zeta_m^p[n + N_m](t)$. The N_m-periodic characteristic sequence of the space ${}^p\mathscr{S}_m$

$$u_m^p[n] \stackrel{\text{def}}{=} \zeta_m^p[n](0) = \sum_{k=0}^{N_m-1} \omega^{-2^m nk} B_m^p(2^m k) = \frac{1}{2^m} \sum_{l=-\infty}^{\infty} \left(\frac{\sin \pi (n/N_m + l)}{\pi (n/N_m + l)} \right)^p \tag{8.5}$$

Proposition 8.1 *The characteristic sequence $u_m^p[n]$ of the space ${}^p\mathscr{S}_m$ can be calculated via the $2^{-m}N-$point DFT of the sampled B-spline $B^p(t) \in {}^p\mathscr{S}$:*

$$u_m^p[n] = \frac{1}{2^m} \sum_{k=0}^{N_m-1} \omega^{-2^m nk} B^p(k). \tag{8.6}$$

The proof is similar to the proof of Proposition 6.1.

A spline $S(t) \in {}^p\mathscr{S}_m$, which is expanded over the B-spline shifts as in Eq. (8.3), can be alternatively represented as

$$S(t) = \frac{2^m}{N} \sum_{n=-N_m/2}^{N_m/2-1} \xi[n]\, \zeta_m^p[n](t), \quad \xi[n] = \hat{q}[n]_m = \sum_{k=0}^{N_m-1} \omega^{-2^m nk} q[k]. \tag{8.7}$$

As in Proposition 4.1, find the norm

$$\|\zeta_m^p[n]\| = \sqrt{N_m\, u_m^{4r}[n]}. \tag{8.8}$$

Thus, the orthonormal basis splines in the space ${}^p\mathscr{S}_m$ are

$$\gamma_m^p[n](t) \stackrel{\text{def}}{=} \sqrt{\frac{1}{N_m\, u_m^{4r}[n]}}\, \zeta_m^p[n](t), \quad n = -N_m/2, \ldots, N_m/2 - 1, \tag{8.9}$$

and a spline $S(t) \in {}^p\mathscr{S}_m$, which is represented as in Eq. (8.3), is expanded

$$S(t) = \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \sigma_{m,0,0}[n]\, \gamma_m^p[n](t), \quad \sigma_{m,0,0}[n] = \sqrt{u_m^{4r}[n]}\, \hat{q}[n]_m. \tag{8.10}$$

Remark 8.1.1 For splines on the initial scale ($m = 0$), the previous notations are retained: $\zeta^p[n](t) \equiv \zeta_0^p[n](t)$, $\gamma^p[n](t) \equiv \gamma_0^p[n](t)$, $u^p[n] \equiv u_0^p[n]$.

The space ${}^p\mathscr{S}_m$ is a subspace of the initial scale space ${}^p\mathscr{S}$. Therefore, any spline $S(t) \in {}^p\mathscr{S}_m,\ m > 0$ represented as in Eq. (8.10) can be expanded over the orthonormal basis of ${}^p\mathscr{S}$:

$$S(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \gamma^p[n](t). \tag{8.11}$$

The set of coordinates $\{\sigma[n]\},\ n = -N/2, \ldots, N/2-1$, is called the SHA spectrum of the spline $S(t)$. Further on, the relation will be established between the coordinates $\{\sigma_{m,0,0}[n]\},\ n = -N_m/2, \ldots, N_m/2 - 1$ of the spline $S(t) \in {}^p\mathscr{S}_m,\ m > 0$ and its SHA spectrum.

8.2 Two-Scale Relations

In the rest of this chapter, we assume that the order of the splines is even: $p = 2r$.

8.2.1 Two-Scale Relations in Spline Spaces

The spline space ${}^{2r}\mathscr{S}_m,\ m > 0$ is a subspace of ${}^{2r}\mathscr{S}_{m-1}$. Therefore, the basis splines $\{\zeta_m^{2r}[n](t)\}$ can be expressed via the splines $\{\zeta_{m-1}^{2r}[n](t)\}$.

Proposition 8.2 *The two-scale relation*

$$\zeta_m^{2r}[n](t) = a_{m-1}[n]\, \zeta_{m-1}^{2r}[n](t) + a_{m-1}[n+N_m]\, \zeta_{m-1}^{2r}[n+N_m](t), \tag{8.12}$$
$$\text{where } a_{m-1}[n] = \frac{1}{2} \cos^{2r} \frac{\pi n}{N_{m-1}},$$

holds for $n = -N_m/2, \ldots, \ldots, N_m/2 - 1$. *In particular,*

$$\zeta_1^{2r}[n](t) = \frac{1}{2} \left(\cos^{2r} \frac{\pi n}{N}\, \zeta^{2r}[n](t) + \sin^{2r} \frac{\pi n}{N}\, \zeta^{2r}[n+N/2](t) \right). \tag{8.13}$$

Proof Keeping in mind that

$$\begin{aligned} \sin^{2r} \frac{\pi n}{N_m} &= \sin^{2r} \frac{2\pi n}{N_{m-1}} = \sin^{2r} \frac{2\pi(n+N_m)}{N_{m-1}} \\ &= \frac{2^{2r}}{2^m} \cos^{2r} \frac{\pi n}{N_{m-1}} \sin^{2r} \frac{\pi n}{N_{m-1}} \\ &= \frac{2^{2r}}{2^m} \cos^{2r} \frac{\pi(n+N_m)}{N_{m-1}} \sin^{2r} \frac{\pi(n+N_m)}{N_{m-1}}, \end{aligned}$$

separate even and odd terms in the Fourier series in Eq. (8.4):

$$\begin{aligned}\zeta_m^{2r}[n](t) &= \frac{\sin^{2r} 2\pi n/N_{m-1}}{2^m} \sum_{l=-\infty}^{\infty} \frac{e^{2\pi i(n/N+2^{-m}l)t}}{(\pi(n/N_m+l))^{2r}} \\ &= \frac{1}{2^m}\cos^{2r}\frac{\pi n}{N_{m-1}} \sin^{2r}\frac{\pi n}{N_{m-1}} \sum_{l=-\infty}^{\infty} \frac{e^{2\pi i(n/N+2^{-m+1}l)t}}{(\pi(n/N_{m-1}+l))^{2r}} \\ &\quad + \frac{1}{2^m}\cos^{2r}\frac{\pi(n+N_m)}{N_{m-1}} \sin^{2r}\frac{\pi(n+N_m)}{N_{m-1}} \sum_{l=-\infty}^{\infty} \frac{e^{2\pi i((n+N_m)/N+2^{-m+1}l)t}}{(\pi((n+N_m)/N_{m-1}+l))^{2r}}.\end{aligned}$$

Comparing the last equation with Eq. (8.4), we see that

$$\zeta_m^{2r}[n](t) = \frac{1}{2}\left(\cos^{2r}\frac{\pi n}{N_{m-1}}\,\zeta_{m-1}^{2r}[n](t) + \cos^{2r}\frac{\pi(n+N_m)}{N_{m-1}}\,\zeta_{m-1}^{2r}[n+N_m](t)\right).$$

■

For further constructions, the two-scale relation between the orthonormal basis splines is needed. It follows directly from Proposition 8.2.

Corollary 8.1 *The two-scale relation*

$$\gamma_m^{2r}[n](t) = s_{m-1}[n]\,\gamma_{m-1}^{2r}[n](t) + s_{m-1}[n+N_m]\,\gamma_{m-1}^{2r}[n+N_m](t), \quad (8.14)$$

$$s_{m-1}[n] \stackrel{\text{def}}{=} \sqrt{\frac{u_{m-1}^{4r}[n]}{2\,u_m^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N_{m-1}}.$$

for the normalized exponential splines holds for $n = -N_m/2, \ldots, N_m/2-1$.

Remark 8.2.1 If $n = -N_m/2, \ldots, -1$, then indices $\bar{n} = n+N_m$ lie in the range $\bar{n} = N_m/2, \ldots, N_m-1$. The splines sequence $\left\{\gamma_{m-1}^{2r}[n](t)\right\}$, $n = -N_m, \ldots, \ldots, N_m-1$ is N_{m-1}-periodic. Therefore, if $n = 0, \ldots, N_m/2-1$, then $\gamma_{m-1}^{2r}[n+N_m](t) = \gamma_{m-1}^{2r}[n-N_m](t)$, where indices $\bar{n} = n-N_m$ lie in the range $\bar{n} = -N_m, \ldots, -N_m/2-1$.

The norms of the splines $\gamma_m^{2r}[n](t)$ and $\gamma_{m-1}^{2r}[n](t)$ are equal to one. Hence,

$$(s_{m-1}[n])^2 + (s_{m-1}[n+N_m])^2 = 1, \quad n = -N_m/2, \ldots, N_m/2-1. \quad (8.15)$$

8.2.2 Orthogonal Complements to Spline Spaces

Denote by ${}^{2r}\mathscr{W}_{m,1}$ the orthogonal complement to the spline space ${}^{2r}\mathscr{S}_m$ in the space ${}^{2r}\mathscr{S}_{m-1}$. We construct an orthonormal basis that characterizes N_m-dimensional space ${}^{2r}\mathscr{W}_{m,1}$. Define the splines for $n = -N_m/2, \ldots, N_m/2-1$

$$\gamma_{m,1}^{2r}[n](t) \stackrel{\text{def}}{=} d_{m-1}[n]\,\gamma_{m-1}^{2r}[n](t) + d_{m-1}[n+N_m]\,\gamma_{m-1}^{2r}[n+N_m](t),$$

$$d_{m-1}[n] \stackrel{\text{def}}{=} \omega^{2^{m-1}n}\,s_{m-1}[n+N_m] = \omega^{2^{m-1}n}\sqrt{\frac{u_{m-1}^{4r}[n+N_m]}{2\,u_m^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N_{m-1}}. \tag{8.16}$$

Proposition 8.3 *The set of splines* $\{\gamma_{m,1}^{2r}[n](t)\},\ n=-N_m/2,\ldots,N_m/2-1$, *forms an orthonormal basis for* ${}^{2r}\mathscr{W}_{m,1}$.

Proof The mutual orthogonality of the splines $\{\gamma_{m,1}^{2r}[n](t)\},\ n=-N_m/2,\ldots,N_m/2-1$, stems from the orthogonality of the "parent" splines $\{\gamma_{m-1}^{2r}[n](t)\}$, where $n=-N_{m-1}/2,\ldots,N_{m-1}/2-1$. If $l\neq n$ then

$$\begin{aligned}
&\left\langle \gamma_{m,1}^{2r}[n],\ \gamma_{m,1}^{2r}[l]\right\rangle = d_{m-1}[n]\,d_{m-1}[l]^*\left\langle \gamma_{m-1}^{2r}[n],\ \gamma_{m-1}^{2r}[l]\right\rangle\\
&+ d_{m-1}[n+N_m]\,d_{m-1}[l]^*\left\langle \gamma_{m-1}^{2r}[n+N_m],\ \gamma_{m-1}^{2r}[l]\right\rangle\\
&+ d_{m-1}[n]\,d_{m-1}[l+N_m]^*\left\langle \gamma_{m-1}^{2r}[n],\ \gamma_{m-1}^{2r}[l+N_m]\right\rangle\\
&+ d_{m-1}[n+N_m]\,d_{m-1}[l+N_m]^*\left\langle \gamma_{m-1}^{2r}[n+N_m],\ \gamma_{m-1}^{2r}[l+N_m]\right\rangle = 0.
\end{aligned}$$

Due to Eq. (8.15), the squared norms

$$\begin{aligned}
\left\|\gamma_{m,1}^{2r}[n]\right\|^2 &= |d_{m-1}[n]|^2\left\|\gamma_{m-1}^{2r}[n]\right\|^2 + |d_{m-1}[n+N_m]|^2\left\|\gamma_{m-1}^{2r}[n+N_m]\right\|^2\\
&= (s_{m-1}[n+N_m])^2 + (s_{m-1}[n])^2 = 1.
\end{aligned}$$

It remains to establish the orthogonality relation between the splines $\gamma_m^{2r}[n]\,(t)$ and $\gamma_{m,1}^{2r}[n]\,(t)$ from the subspaces ${}^{2r}\mathscr{S}_m$ and ${}^{2r}\mathscr{W}_{m,1}$ of the space ${}^{2r}\mathscr{S}_{m-1}$, respectively. From Eqs. (8.14) and (8.16), we have:

$$\begin{aligned}
&\left\langle \gamma_m^{2r}[n],\ \gamma_{m,1}^{2r}[n]\right\rangle = s_{m-1}[n]\,d_{m-1}[n]^* + s_{m-1}[n+N_m]\,d_{m-1}[n+N_m]^*\\
&= \omega^{-2^{m-1}n}\,(s_{m-1}[n]\,s_{m-1}[n+N_m] - s_{m-1}[n+N_m]\,s_{m-1}[n]) = 0.
\end{aligned} \tag{8.17}$$

■

Remark 8.2.2 Note that the union $\left\{\gamma_m^{2r}[n](t)\right\}\bigcup\left\{\gamma_{m,1}^{2r}[n](t)\right\},\ n=-N_m/2,\ldots,N_m/2-1$, forms an orthonormal basis for the entire space ${}^{2r}\mathscr{S}_{m-1}$.

Proposition 8.4 *The splines* $\gamma_m^{2r}[n]\,(t)$ *and* $\gamma_{m,1}^{2r}[n]\,(t)$ *are the eigenvectors of the shift operator*

$$\gamma_m^{2r}[n]\left(t+2^m k\right) = \omega^{2^m nk}\,\gamma_m^{2r}[n]\,(t)\,, \tag{8.18}$$

$$\gamma_{m,1}^{2r}[n]\left(t+2^m k\right) = \omega^{2^m nk}\,\gamma_{m,1}^{2r}[n]\,(t)\,,\quad k=0,\ldots,N_m-1. \tag{8.19}$$

Proof At the initial scale, Eq. (4.20) implies that $\gamma^{2r}[n]\,(t+k) = \omega^{nk}\gamma^{2r}[n](t)$. At the first scale for $l = 0, 1$

$$\begin{aligned}\gamma_{1,0}^{2r}[n]\,(t+2k) &= s_0[n]\,\gamma^{2r}[n]\,(t+2k) + s_0[n+N/2]\,\gamma^{2r}[n+N/2]\,(t+2k)\\ &= \omega^{2kn}\,\gamma_{1,0}^{2r}[n](t).\end{aligned}$$

For $m > 1$, Eq. (8.19) is derived by induction. The proof for the splines $\gamma_{m,1}^{2r}[n]$ is similar. ■

8.2.3 Refined Split of the Spline Space into Orthogonal Subspaces

The initial spline space ${}^{2r}\mathscr{S}$, which can be regarded as the linear span of the orthonormal basis splines $\left\{\gamma^{2r}[n](t)\right\}$, $n = -N/2, \ldots, N/2-1$, is split into the mutually orthogonal subspaces of the first decomposition level ${}^{2r}\mathscr{S} = {}^{2r}\mathscr{S}_1 \bigoplus {}^{2r}\mathscr{W}_{1,1}$. The subspaces ${}^{2r}\mathscr{S}_1$ and ${}^{2r}\mathscr{W}_{1,1}$ are the spans of the orthonormal sets $\left\{\gamma_{1,0}^{2r}[n](t)\right\}$ and $\left\{\gamma_{1,1}^{2r}[n](t)\right\}$, respectively, where $n = -N_1/2, \ldots, N_1/2-1$.

We apply a similar procedure to either of the "offspring" subspaces. The spline subspace is split ${}^{2r}\mathscr{S}_1 = {}^{2r}\mathscr{S}_2 \bigoplus {}^{2r}\mathscr{W}_{2,1}$. The orthonormal bases of these subspaces are $\left\{\gamma_2^{2r}[n](t)\right\}$ and $\left\{\gamma_{2,1}^{2r}[n](t)\right\}$, $n = -N_2/2, \ldots, N_2/2-1$, respectively.

In order to split the subspace ${}^{2r}\mathscr{W}_{1,1}$, we define two sets $\left\{\gamma_{2,2}^{2r}[n](t)\right\}$ and $\left\{\gamma_{2,3}^{2r}[n](t)\right\}$, $n = -N_2/2, \ldots, N_2/2-1$.

$$\begin{aligned}\gamma_{2,2}^{2r}[n](t) &\stackrel{\text{def}}{=} d_1[n]\,\gamma_{1,1}^{2r}[n](t) + d_1[n+N_2]\,\gamma_{1,1}^{2r}[n+N_2](t),\\ \gamma_{2,3}^{2r}[n](t) &\stackrel{\text{def}}{=} s_1[n]\,\gamma_{1,1}^{2r}[n](t) + s_1[n+N_2]\,\gamma_{1,1}^{2r}[n+N_2](t),\\ s_1[n] &\stackrel{\text{def}}{=} \sqrt{\frac{u_1^{4r}[n]}{2\,u_2^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N_1},\quad d_1[n] \stackrel{\text{def}}{=} \omega^{2^2 n}\sqrt{\frac{u_1^{4r}[n+N_2]}{2\,u_2^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N_1}.\end{aligned}$$

Similarly to Proposition 8.3, it can be verified that the splines $\left\{\gamma_2^{2r}[n](t)\right\}$, and $\left\{\gamma_{2,p}^{2r}[n](t)\right\}$, $p = 1, 2, 3$, where $n = -N_2/2, \ldots, N_2/2-1$, have norms equal to one. They are mutually orthogonal within each set and any spline belonging to a certain set is orthogonal to all the splines from the remaining three sets.

These orthonormal sets produce a natural split of the space ${}^{2r}\mathscr{W}_{1,1}$ into orthogonal subspaces: ${}^{2r}\mathscr{W}_{1,1} = {}^{2r}\mathscr{W}_{2,2} \bigoplus {}^{2r}\mathscr{W}_{2,3}$, where the subspaces ${}^{2r}\mathscr{W}_{2,p}$, $p = 2, 3$, consist of splines represented as

$$S(t) = \frac{1}{\sqrt{N_2}} \sum_{n=-N_2/2}^{N_2/2-1} \sigma_{2,p}[n]\, \gamma_{2,p}^{2r}[n](t), \quad p = 2, 3.$$

Thus, the space ${}^{2r}\mathscr{S}$ is split into four orthogonal subspaces of the second decomposition level

$${}^{2r}\mathscr{S} = {}^{2r}\mathscr{S}_2 \bigoplus {}^{2r}\mathscr{W}_{2,1} \bigoplus {}^{2r}\mathscr{W}_{2,1} \bigoplus {}^{2r}\mathscr{W}_{2,2} \bigoplus {}^{2r}\mathscr{W}_{2,3} \tag{8.20}$$

and the union

$$\left\{\gamma_2^{2r}[n](t)\right\} \bigcup \left\{\gamma_{2,1}^{2r}[n](t)\right\} \bigcup \left\{\gamma_{2,2}^{2r}[n](t)\right\} \bigcup \left\{\gamma_{2,3}^{2r}[n](t)\right\},\ n = 0, \ldots, N_2 - 1,$$

forms an orthonormal basis for the space ${}^{2r}\mathscr{S}$.

For uniformity, denote the space ${}^{2r}\mathscr{S}_m$ *as* ${}^{2r}\mathscr{W}_{m,0}$ *and, respectively, the splines* $\gamma_m^{2r}[n](t)$ *as* $\gamma_{m,0}^{2r}[n](t)$.

Then, Eq. (8.20) can be rewritten as ${}^{2r}\mathscr{S} = \bigoplus_{p=0}^{3} {}^{2r}\mathscr{W}_{2,p}$.

Such a decomposition scheme is expanded to the third level producing eight mutually orthogonal subspaces and so on. In general, we assume that the set of exponential splines $\left\{\gamma_{m-1,l}^{2r}[n](t)\right\},\ n = 0, \ldots, N_{m-1} - 1$, forms an orthonormal basis of the space ${}^{2r}\mathscr{W}_{m-1,l}$. We construct a new orthonormal basis that consists of two different blocks, which are orthogonal to each other, using the coefficients $s_{m-1}[n]$ and $d_{m-1}[n]$ defined in Eqs. (8.14) and (8.16).

If l is even, then

$$\begin{aligned} g_{m,2l}^{2r}[n](t) &= s_{m-1}[n]\, \gamma_{m-1,l}^{2r}[n](t) + s_{m-1}[n+N_m]\, \gamma_{m-1,l}^{2r}[n+N_m](t), \\ \gamma_{m,2l+1}^{2r}[n](t) &= d_{m-1}[n]\, \gamma_{m-1,l}^{2r}[n](t) + d_{m-1}[n+N_m]\, \gamma_{m-1,l}^{2r}[n+N_m](t). \end{aligned} \tag{8.21}$$

If l is odd then

$$\begin{aligned} \gamma_{m,2l}^{2r}[n](t) &= d_{m-1}[n]\, \gamma_{m-1,l}^{2r}[n](t) + d_{m-1}[n+N_m]\, \gamma_{m-1,l}^{2r}[n+N_m](t), \\ \gamma_{m,2l+1}^{2r}[n](t) &= s_{m-1}[n]\, \gamma_{m-1,l}^{2r}[n](t) + s_{m-1}[n+N_m]\, \gamma_{m-1,l}^{2r}[n+N_m](t). \end{aligned} \tag{8.22}$$

Then, the spaces ${}^{2r}\mathscr{W}_{m,2l}$ and ${}^{2r}\mathscr{W}_{m,2l+1}$, which produce an orthogonal split of the space ${}^{2r}\mathscr{W}_{m-1,l}$ are formed as linear spans of the orthonormal systems $\left\{\gamma_{m,2l}^{2r}[n](t)\right\}$ and $\left\{\gamma_{m,2l+1}^{2r}[n](t)\right\}$, respectively, where $n = -N_m/2, \ldots, N_m/2 - 1$. Similarly to Proposition 8.4, the following shift property can be proved.

Proposition 8.5 *The splines* $\gamma_{m,l}^{2r}[n]\,(t)$ *are the eigenvectors of the shift operator*

$$\gamma_{m,l}^{2r}[n]\left(t + 2^m k\right) = \omega^{2^m nk}\, \gamma_{m,l}^{2r}[n]\,(t), \quad k = 0, \ldots, N_m - 1. \tag{8.23}$$

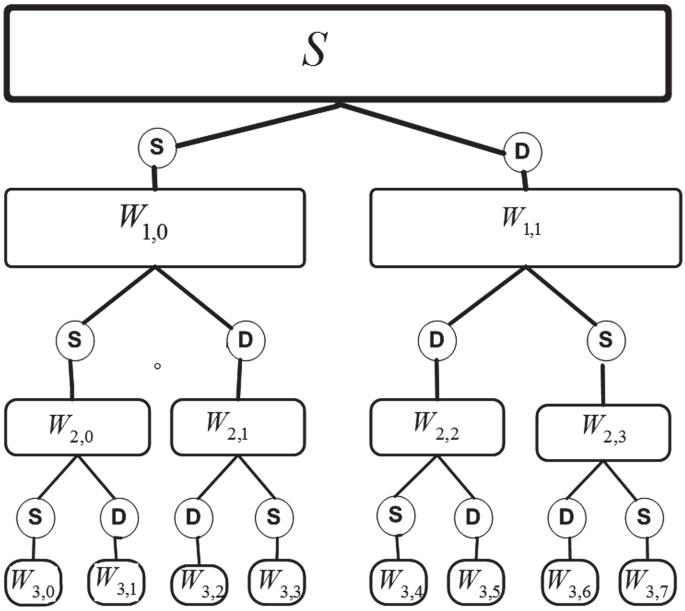


Fig. 8.1 Diagram of 3-level orthogonal decomposition of the spline space

Note that application of the coefficients $s_{m-1}[n]$ and $d_{m-1}[n]$ in Eqs. (8.21) and (8.22) is interchanged (see Remark 8.3.1).

Consequently, the spline space ${}^{2r}\mathscr{S}$ can be decomposed into a collection of orthogonal sums:

$$ {}^{2r}\mathscr{S} = {}^{2r}\mathscr{W}_{1,0} \bigoplus {}^{2r}\mathscr{W}_{1,1} = \bigoplus_{l=0}^{3} {}^{2r}\mathscr{W}_{2,l} = \ldots = \bigoplus_{l=0}^{2^m-1} {}^{2r}\mathscr{W}_{m,l}. \tag{8.24} $$

The diagram in Fig. 8.1 illustrates the 3-level orthogonal decomposition of the spline space ${}^{2r}\mathscr{S}$. A circle with the letter **s** inside means application of the coefficients $s_{m-1}[n]$ to the basis splines, while a circle with the letter **d** inside means application of the coefficients $d_{m-1}[n]$.

Remark 8.2.3 Once the collection of different orthogonal subspaces of ${}^{2r}\mathscr{S}$ is available, different configurations of the split of ${}^{2r}\mathscr{S}$ are possible. For example, when 3-level decomposition is performed, the splits

$$ {}^{2r}\mathscr{S} = {}^{2r}\mathscr{W}_{1,0} \bigoplus {}^{2r}\mathscr{W}_{2,2} \bigoplus {}^{2r}\mathscr{W}_{3,7} \bigoplus {}^{2r}\mathscr{W}_{3,7} \text{ or} \tag{8.25} $$

$$ {}^{2r}\mathscr{S} = {}^{2r}\mathscr{W}_{2,0} \bigoplus {}^{2r}\mathscr{W}_{3,2} \bigoplus {}^{2r}\mathscr{W}_{3,1} \bigoplus {}^{2r}\mathscr{W}_{1,1} \tag{8.26} $$

and many others are possible. An optimal split is implemented via the Best Basis algorithm [4, 9, 10, 14], which will be discussed in Sect. 8.4.4. If the decomposition (8.25) takes place, then the orthonormal basis of ${}^{2r}\mathscr{S}$ is formed by the exponential splines

$$\mathscr{B}_1 \stackrel{\text{def}}{=} \left\{\gamma_{1,0}^{2r}[n](t)\right\}_{n=-N/4}^{N/4-1} \bigcup \left\{\gamma_{2,2}^{2r}[n](t)\right\}_{n=-N/8}^{N/8-1} \bigcup \left\{\gamma_{3,7}^{2r}[n](t)\right\}_{n=-N/16}^{N/16-1} \bigcup \left\{\gamma_{3,8}^{2r}[n](t)\right\}_{n=-N/16}^{N/16-1}.$$

In the case (8.26), the orthonormal basis of ${}^{2r}\mathscr{S}$ is

$$\mathscr{B}_2 \stackrel{\text{def}}{=} \left\{\gamma_{2.0}^{2r}[n](t)\right\}_{n=-N/8}^{N/8-1} \bigcup \left\{\gamma_{1,1}^{2r}[n](t)\right\}_{n=-N/4}^{N/4-1} \bigcup \left\{\gamma_{3,2}^{2r}[n](t)\right\}_{n=-N/16}^{N/16-1} \bigcup \left\{\gamma_{3,1}^{2r}[n](t)\right\}_{n=-N/16}^{N/16-1}.$$

8.3 Spline Wavelet Packet Transforms

In this section we discuss transforms of the splines coordinates while changing from one orthonormal basis to another.

8.3.1 Transform Matrices

Define the matrix

$$\begin{aligned}\mathbf{A}_{m-1}[n] &\stackrel{\text{def}}{=} \begin{pmatrix} s_{m-1}[n] & s_{m-1}[n+N_m] \\ d_{m-1}[n] & d_{m-1}[n+N_m] \end{pmatrix} \\ &= \begin{pmatrix} s_{m-1}[n] & s_{m-1}[n+N_m] \\ \omega^{2^{m-1}n}\, s_{m-1}[n+N_m] & -\omega^{2^{m-1}n}\, s_{m-1}[n] \end{pmatrix}, \\ s_{m-1}[n] &= \sqrt{\frac{u_{m-1}^{4r}[n]}{2\,u_m^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N_{m-1}}, \quad N_m = \frac{N}{2^m}.\end{aligned} \tag{8.27}$$

The conjugate transpose of the matrix $\mathbf{A}_{m-1}[n]$ denote by

$$\mathbf{A}_{m-1}^{\dagger}[n] \stackrel{\text{def}}{=} \begin{pmatrix} s_{m-1}[n] & d_{m-1}^{*}[n] \\ s_{m-1}[n+N_m] & d_{m-1}^{*}[n+N_m] \end{pmatrix}.$$

Proposition 8.6 *With any natural m, the matrices* $\mathbf{A}_m[n]$ *are unitary, that is*

$$\mathbf{A}_{m-1}[n] \cdot \mathbf{A}_{m-1}^{\dagger}[n] = \mathbf{A}_{m-1}^{\dagger}[n] \cdot \mathbf{A}_{m-1}[n] = \mathbf{I},$$

where $\mathbf{I}$ *denotes the* 2×2 *identity matrix.*

Proof Equation (8.15) implies that

$$\begin{aligned}\mathbf{A}_{m-1}[n] \cdot \mathbf{A}_{m-1}^{\dagger}[n] &= \begin{pmatrix} s_{m-1}[n] & s_{m-1}[n+N_m] \\ \omega^{2m-1n}\, s_{m-1}[n+N_m] & -\omega^{2^{m-1}n}\, s_{m-1}[n] \end{pmatrix} \\ &\quad \times \begin{pmatrix} s_{m-1}[n] & \omega^{-2^{m-1}n}\, s_{m-1}[n+N_m] \\ s_{m-1}[n+N_m] & -\omega^{-2^{m-1}n}\, s_{m-1}[n] \end{pmatrix} \\ &= \begin{pmatrix} \big(s_{m-1}[n]\big)^2 + \big(s_{m-1}[n+N_m]\big)^2 & 0 \\ 0 & \big(s_{m-1}[n]\big)^2 + \big(s_{m-1}[n+N_m]\big)^2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{I}.\end{aligned}$$

Similarly, the relation $\mathbf{A}_{m-1}^{\dagger}[n] \cdot \mathbf{A}_{m-1}[n] = \mathbf{I}$ is verified. ■

Corollary 8.2 *The inverse matrix*

$$\mathbf{A}_{m-1}^{-1}[n] = \mathbf{A}_{m-1}^{\dagger}[n] = \begin{pmatrix} s_{m-1}[n] & \omega^{-2^{m-1}n}\, s_{m-1}[n+N_m] \\ s_{m-1}[n+N_m] & -\omega^{-2^{m-1}n}\, s_{m-1}[n] \end{pmatrix}.$$

8.3.2 One-Level Transforms

Let a spline $S(t) \in {}^{2r}\mathscr{S}$ be represented by the orthonormal basis splines

$$S(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n] \gamma^{2r}[n](t). \tag{8.28}$$

8.3.2.1 Transforms of the Splines' Coordinates

The sequence $\{\sigma[n]\}$, $n = -N/2, \ldots, N/2-1$, is the SHA spectrum of the spline $S(t)$. The space ${}^{2r}\mathscr{S}$ is the orthogonal sum of the subspaces ${}^{2r}\mathscr{S}_1 = {}^{2r}\mathscr{W}_{1,0}$ and ${}^{2r}\mathscr{W}_{1,1}$ whose orthonormal bases are $\left\{\gamma_{1,0}^{2r}[n](t)\right\}$ and $\left\{\gamma_{1,1}^{2r}[n](t)\right\}$, respectively, where $n = -N_1/2, \ldots, N_1/2-1$ and $N_1 = N/2$. Thus, $S(t)$ can be represented as the sum of its orthogonal projections onto the subspaces ${}^{2r}\mathscr{W}_{1,0}$ and ${}^{2r}\mathscr{W}_{1,1}$: $S(t) = S_{1,0}(t) + S_{1,1}(t)$, where

$$S_{1,p}(t) \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \sigma_{1,p}[n]\gamma_{1,p}^{2r}[n](t), \quad p = 0, 1. \tag{8.29}$$

The orthonormality of the basis splines implies that the coordinates

$$\sigma[n] = \sqrt{N}\left\langle S, \gamma^{2r}[n]\right\rangle, \quad \sigma_{1,p}[n] = \sqrt{N/2}\left\langle S, \gamma_{1,p}^{2r}[n]\right\rangle.$$

Using the two-scale relations (8.14) and (8.16), we derive for $n = -N/4, \ldots, N/4 - 1$

$$\sigma_{1,0}[n] = \sqrt{N/2}\left\langle S, \gamma_{1,0}^{2r}[n]\right\rangle = \sqrt{N/2}\left\langle S, \left(s_0[n]\,\gamma^{2r}[n] + s_0[n+N/2]\,\gamma^{2r}[n+N/2]\right)\right\rangle$$
$$= \sqrt{\frac{1}{2}}\,\left(s_0[n]\,\sigma[n] + s_0[n+N/2]\,\sigma[n+N/2]\right), \tag{8.30}$$

$$\sigma_{1,1}[n] = \sqrt{\frac{1}{2}}\,\left(d_0[n]\,\sigma[n](t) + d_0[n+N/2]\,\sigma[n+N/2]\right),$$
$$s_0[n] = \sqrt{\frac{u^{4r}[n]}{2\,u_1^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N},$$
$$d_0[n] = \omega^n \sqrt{\frac{u^{4r}[n+N/2]}{2\,u_1^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N} = \omega^n\, s_0[n+N/2]. \tag{8.31}$$

Equations (8.30) and (8.31) can be presented in a matrix form

$$\begin{pmatrix}\sigma_{1,0}[n]\\ \sigma_{1,1}[n]\end{pmatrix} = \sqrt{\frac{1}{2}}\mathbf{A}_0[n]\cdot\begin{pmatrix}\sigma[n]\\ \sigma[n+N/2]\end{pmatrix}, \quad n = -N/4, \ldots, N/4-1. \tag{8.32}$$

The inverse relation is

$$\begin{pmatrix}\sigma[n]\\ \sigma[n+N/2]\end{pmatrix} = \sqrt{2}\mathbf{A}_0^{\dagger}[n]\cdot\begin{pmatrix}\sigma_{1,0}[n]\\ \sigma_{1,1}[n]\end{pmatrix}$$
$$= \sqrt{2}\begin{pmatrix}s_0[n] & d_0^*[n]\\ s_0[n+N/2] & d_0^*[n+N/2]\end{pmatrix}\cdot\begin{pmatrix}\sigma_{1,0}[n]\\ \sigma_{1,1}[n]\end{pmatrix}, \quad n = -N/2, \ldots, N/2-1. \tag{8.33}$$

It follows from Eq. (8.33) that, once the orthogonal projections of the spline $S(t)$ onto the subspaces ${}^{2r}\mathscr{W}_{1,0}$ and ${}^{2r}\mathscr{W}_{1,1}$ are available (see Eq. (8.29)) then the spline is reconstructed as

$$S(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\gamma^{2r}[n](t),$$
$$\sigma[n] = \sqrt{2}\left(s_0[n]\,\sigma_{1,0}[n] + d_0^*[n]\,\sigma_{1,1}[n]\right)$$

$$= \frac{1}{\sqrt{u_1^{4r}[n]}} \left(\sqrt{u^{4r}[n]}\, \cos^{2r} \frac{\pi n}{N}\, \sigma_{1,0}[n] + \omega^{-n} \sqrt{u^{4r}[n+N/2]}\, \sin^{2r} \frac{\pi n}{N}\, \sigma_{1,1}[n] \right). \tag{8.34}$$

In particular, the splines $S_{1,0}(t) \in {}^{2r}\mathscr{W}_{1,0}$ and $S_{1,1}(t) \in {}^{2r}\mathscr{W}_{1,1}$ are represented as

$$S_{1,0}(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \varsigma_{1,0}[n]\, \gamma^{2r}[n](t), \quad S_{1,1}(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \varsigma_{1,1}[n]\, \gamma^{2r}[n](t),$$

$$\varsigma_{1,0}[n] \stackrel{\text{def}}{=} \sqrt{2} s_0[n]\, \sigma_{1,0}[n] = \sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}}\, \cos^{2r} \frac{\pi n}{N}\, \sigma_1[n],$$

$$\varsigma_{1,1}[n] \stackrel{\text{def}}{=} \sqrt{2} d_0^*[n]\, \sigma_{1,1}[n] = \omega^{-n} \sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\, \sin^{2r} \frac{\pi n}{N}\, \sigma_{1,1}[n]. \tag{8.35}$$

The sequences $\left\{\varsigma_{1,0}[n]\right\}$, $n = -N/2, \ldots, N/2-1$ and $\left\{\varsigma_{1,1}[n]\right\}$, $n = -N/2, \ldots, N/2-1$ form the SHA spectra of the splines $S_{1,0}(t) \in {}^{2r}\mathscr{W}_{1,0}$ and $S_{1,1}(t) \in {}^{2r}\mathscr{W}_{1,1}$, respectively. Note that the sequences $\left\{u_1^{4r}[n]\right\}$, $\left\{\sigma_{1,0}[n]\right\}$ and $\left\{\sigma_{1,1}[n]\right\}$ are $N/2$-periodic.

8.3.2.2 Remarks on the SHA Spectra Transforms

Assume a spline

$$S(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n] \gamma^{2r}[n](t).$$

It follows from the discussion in Remark 8.2.1 that, if indices ν lie in the range $-N/4, \ldots, N/4-1$, then $\{\sigma[n]\}$ compose the low-frequency part of the SHA spectrum of $S(t)$. The set

$$\{\sigma[n+N/2]\}_{n=-N/4}^{N/4-1} = \{\sigma[n]\}_{n=-N/2}^{-N/4-1} \bigcup \{\sigma[n]\}_{n=N/4}^{N/2-1}$$

composes the high-frequency part of the spline's SHA spectrum.

Equation (8.35) implies that the SHA spectrum $\varsigma_{1,0}[n]$ of the splines $S_{1,0}(t) \in {}^{2r}\mathscr{S}$ is the product of the N-periodic sequence $s_0[n]$ and the $N/2$-periodic sequence $\sqrt{2}\,\sigma_{1,0}[n]$. Due to Proposition 4.36, the spline $S_{1,0}(t)$ is the result of s-filtering the spline

$$\tilde{S}_{1,0}(t) = \sqrt{\frac{2}{N}} \sum_{n=-N/2}^{N/2-1} \sigma_{1,0}[n]\, \gamma^{2r}[n](t)$$
$$= \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \sigma_{1,0}[n] \left(\gamma^{2r}[n](t) + \gamma^{2r}\left[n + \frac{N}{2} \right](t) \right)$$

with the s-filter $\mathbf{s}_{0,0}$, whose frequency response is $s_0[n]$. Respectively, the spline $S_{1,0}(t) \in {}^{2r}\mathscr{S}$ is the result of s-filtering the spline

$$\tilde{S}_{1,1}(t) = \sqrt{\frac{2}{N}} \sum_{n=-N/2}^{N/2-1} \sigma_{1,1}[n]\, \gamma^{2r}[n](t)$$
$$= \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \sigma_{1,1}[n] \left(\gamma^{2r}[n](t) + \gamma^{2r}\left[n + \frac{N}{2} \right](t) \right)$$

with the s-filter $\mathbf{s}_{0,1}$, whose frequency response is $d_0^*[n] = \omega^{-n} s_0[n + N/2]$.

Proposition 8.7 *The sequence* $\{s_0[n]\}_{n=-N/2}^{N/2-1}$ *presents the frequency response of a low-pass p-filter, while the sequence* $\{d_0[n] = \omega^n s_0[n + N/2]\}_{n=-N/2}^{N/2-1}$ *presents the frequency response of a high-pass p-filter.*

Proof Equation (8.5) implies that the characteristic sequences of the spaces ${}^{4r}\mathscr{S}$ and ${}^{4r}\mathscr{W}_{1,0}$ are

$$u^{4r}[n] = \sum_{l \in \mathbb{Z}} \left(\frac{\sin \pi(n/N + l)}{\pi(n/N + l)} \right)^{4r}$$
$$= \left(\frac{\sin \pi(n/N)}{\pi n/N} \right)^{4r} \left(1 + \frac{1}{(1 - N/n)^{4r}} + \frac{1}{(1 + N/n)^{4r}} \right) + O(N^{-4r}), \quad (8.36)$$

$$u_1^{4r}[n] = \frac{1}{2} \sum_{l \in \mathbb{Z}} \left(\frac{\sin \pi(2n/N + l)}{\pi(2n/N + l)} \right)^{4r}$$
$$= \frac{1}{2} \left(\frac{\sin 2\pi(n/N)}{2\pi n/N} \right)^{4r} \left(1 + \frac{1}{(1 - 2N/n)^{4r}} + \frac{1}{(1 + 2N/n)^{4r}} \right) + O(N^{-4r}). \quad (8.37)$$

If $|n| \ll N$ then

$$u^{4r}[n] = \left(\frac{\sin \pi(n/N)}{\pi n/N} \right)^{4r} + O(N^{-4r}), \; u_1^{4r}[n] = \frac{1}{2} \left(\frac{\sin 2\pi(n/N)}{2\pi n/N} \right)^{4r} + O(N^{-4r})$$

and we have

$$s_0[n] = \sqrt{\frac{u^{4r}[n]}{2\,u_1^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N} \approx \left(\frac{2\pi n/N\ \sin\pi(n/N)}{\pi n/N\ \sin 2\pi(n/N)}\right)^{2r}\cos^{2r}\frac{\pi n}{N}$$
$$= \left(\frac{2\sin\pi(n/N)}{2\sin\pi(n/N)\ \cos\pi(n/N)}\right)^{2r}\cos^{2r}\frac{\pi n}{N} = 1.$$

Note that this "flat" interval is the longer, the higher spline's order is, due to the faster decay of the remainder terms in the expressions (8.36) and (8.36). When n is close to $N/2$,

$$u^{4r}[n] \approx \sin^{4r}\frac{\pi n}{N}\left(\frac{1}{(\pi(n/N))^{4r}} + \frac{1}{(\pi(n/N)-1)^{4r}}\right) + O(N^{-4r}).$$

Thus, $u^{4r}[n]$ is decaying as $n \to N/2$ and $\lim_{n\to N/2} u^{4r}[n] \approx 2(\pi/2)^{-4r}$.

On the other hand, when n is close to $N/2$, $u_1^{4r}[n]$ can be represented as

$$u_1^{4r}[n] = \frac{1}{2}\left(\frac{\sin\pi(2n/N-1)}{\pi(2n/N-1)}\right)^{4r} + \frac{1}{2}\sin^{4r}\pi(2n/N-1)\sum_{l\in\mathbb{Z}\setminus -1}\left(\frac{1}{\pi(2n/N+l)}\right)^{4r}.$$

Thus, $u_1^{4r}[n]$ is growing as $n \to N/2$ and $u_1^{4r}[N/2] = 1/2$. Consequently, the fraction $\sqrt{u^{4r}[n]/2\,u_1^{4r}[n]}$ is decaying as $n \to N/2$ thus enhancing the decay of $\cos^{2r}\pi n/N$. The sequence $s_0[n]$ is fast decaying as $n \to N/2$ and $s_0[N/2] = 0$. Recall that it is symmetric about zero. Therefore, it is the frequency response of a low-pass p-filter, while $\{d_0[n] = \omega^n s_0[n+N/2]\}_{n=-N/2}^{N/2-1}$ is the frequency response of a high-pass p-filter. ■

We denote by $\mathbf{s}_{0,0}$ the p-filter whose frequency response is $\{s_0[n]\}$ and by $\mathbf{s}_{0,1}$ the p-filter whose frequency response is $\{d_0^*[n]\}$, where $n = -N/2, \ldots, N/2-1$.

Figure 8.2 displays the magnitude responses of the p-filters $\mathbf{s}_{0,0}$ and $\mathbf{s}_{0,1}$ for the splines of orders 2, 4, 6 and 8. Observe that the magnitude responses of the p-filters tend to rectangles when the splines orders increase.

Equation (8.30) means that the coefficients $\{\sigma_{1,0}[n]\}$, $n = -N/4, \ldots, N/4-1$, are derived mainly from the low-frequency components $\{\sigma[n]\}$, $n = -N/4, \ldots, N/4-1$, of the SHA spectra of $S(t)$, while the high-frequency components $\{\sigma[n]\}$, $n = (-N/2, \ldots, -N/4-1)\bigcup(N/4, \ldots, N/2-1)$ are suppressed, especially for the splines of higher order. It is seen from Eq. (8.31) that, for the coefficients $\{\sigma_{1,1}[n]\}$, $n = -N/4, \ldots, N/4-1$, the situation is just opposite.

As for the reconstruction of $S(t)$ from the orthogonal projections, Eq. (8.34) demonstrates that the low-frequency components $\{\sigma[n]\}$, $n = -N/4, \ldots, N/4-1$, of the SHA spectra are restored mainly from $\{\sigma_{1,0}[n]\}$, $n = -N/4, \ldots, N/4-1$, with minimal participation of $\{\sigma_{1,1}[n]\}$, $n = -N/4, \ldots, N/4-1$, and vice versa for the high-frequency components $\{\sigma[n]\}$, $n = (-N/2, \ldots, -N/4-1)\bigcup(N/4, \ldots, N/2-1)$. The plots were produced by the MATLAB code `S_filters.m`.

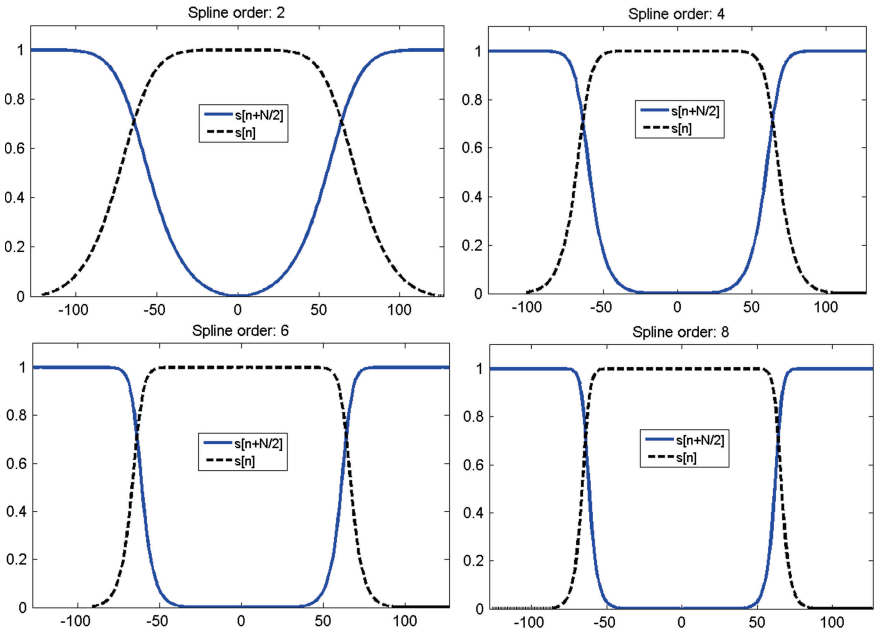


Fig. 8.2 The magnitude responses $\mathbf{s}_{0,0}$ (*dashed curves*) and $\left|\mathbf{s}_{0,1}\right| = \{s_0[n + N/2]\}_{n=-N/2}^{N/2-1}$ (*solid curves*) for the splines of orders 2, 4, 6 and 8, $N = 256$.

8.3.2.3 Periodic Battle–Lemarié Wavelets

We denote

$$\varphi^{2r}(t) \stackrel{\text{def}}{=} \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \gamma^{2r}[n](t), \tag{8.38}$$

These splines, whose SHA spectrum $\nu[n] = 1,\ n = -N/2, \ldots, N/2 - 1$, were introduced in Section 4.1.4, and the following property was established.

Proposition 8.8 *The translations* $\left\{\varphi^{2r}(t-k)\right\},\ k = 0, \ldots, N-1$, *of the splines* $\varphi^{2r}(t)$ *form an orthonormal basis of the space* ${}^{2r}\mathscr{S}$.

The splines $\varphi^{2r}(t)$ are real-valued and can be produced by periodization of the non-periodic self-dual spline generators (Battle–Lemarié father wavelets) [2, 6]. Thus, it is proper to call them the periodic Battle-Lemarié father wavelets.

Periodic Battle-Lemarié wavelets of the first decomposition level
Define the splines

$$\psi_{1,0}^{2r}(t) \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \gamma_{1,0}^{2r}[n](t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \nu_{1,0}[n]\gamma^{2r}[n](t), \quad (8.39)$$

$$\psi_{1,1}^{2r}(t) \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \gamma_{1,1}^{2r}[n](t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \nu_{1,1}[n]\gamma^{2r}[n](t). \quad (8.40)$$

Due to Eq. (8.35), their SHA spectra are

$$\nu_{1,0}[n] = \sqrt{2}s_0[n] = \sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}}\, \cos^{2r}\frac{\pi n}{N}, \quad (8.41)$$

$$\nu_{1,1}[n] = \sqrt{2}d_0^*[n] = \omega^{-n}\sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\, \sin^{2r}\frac{\pi n}{N},$$

respectively. It follows from the shift property Eq. (4.20) of the exponential splines that the spline $\psi_{1,1}^{2r}(t+1)$ has a real-valued SHA spectrum:

$$\psi_{1,1}^{2r}(t+1) = \sqrt{\frac{1}{N}} \sum_{n=1}^{N} \bar{\nu}_{1,1}[n]\gamma^{2r}[n](t), \quad (8.42)$$

$$\bar{\nu}_{1,1}[n] = \sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\, \sin^{2r}\frac{\pi n}{N} = \nu_{1,0}[n+N/2].$$

The splines $\psi_{1,1}^{2r}(t)$ are called the periodic Battle–Lemarié mother wavelets of the first decomposition level. They can be produced by periodization of the non-periodic self-dual (Battle–Lemarié) mother wavelets [2, 6]. The splines $\psi_{1,0}^{2r}(t)$ are the periodic Battle–Lemarié father wavelets of the first level.

Equation (4.30) implies that the grid samples of the splines are

$$\psi_{1,0}^{2r}(k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \nu_{1,0}[n]\,\frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\,\omega^{kn} = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{u^{2r}[n]}{\sqrt{u_1^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N}\,\omega^{kn},$$

$$\psi_{1,1}^{2r}(k) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \nu_{1,1}[n]\,\frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\,\omega^{kn}$$

$$= \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \frac{u^{2r}[n]}{u_1^{4r}[n]}\sqrt{u^{4r}[n+N/2]}\,\sin^{2r}\frac{\pi n}{N}\,\omega^{(k-1)n}.$$

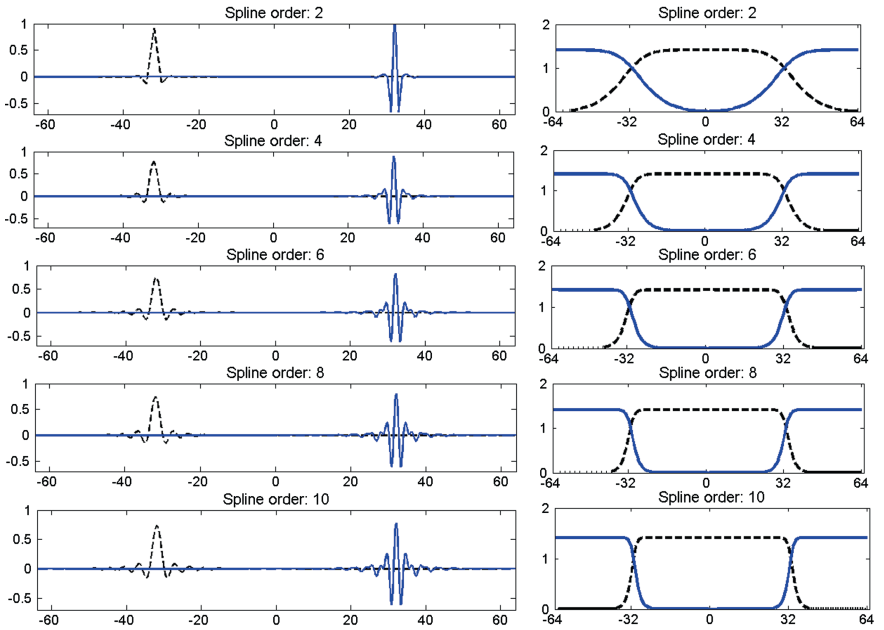


Fig. 8.3 *Left* Periodic Battle–Lemarié wavelets $\psi_{1,0}^{2r}(t)$ (*dashed curves*) and $\psi_{1,1}^{2r}(t)$ (*solid curves*), $r = 1, 2, 3, 4, 5$, of the first decomposition level. *Right* Their SHA spectra $\nu_{1,0}[n]$ (*dashed curves*), and $\bar{\nu}_{1,1}[n]$ (*solid curves*), $N = 128$

Proposition 8.9 *Translations* $\left\{\psi_{1,0}^{2r}(t-2k)\right\}$ *and* $\left\{\psi_{1,1}^{2r}(t-2k)\right\}$, $k = 0, \ldots, N/2-1$, *of the first level wavelets form orthonormal bases of the subspaces* ${}^{2r}\mathscr{W}_{1,0}$ *and* ${}^{2r}\mathscr{W}_{1,1}$, *respectively.*

Proof The orthonormality (Proposition 8.3) and the shift property (8.23) of the exponential splines $\gamma_{1,1}^{2r}[n](t)$ imply that the inner product

$$\int_0^N \psi_{1,1}^{2r}(t-2k)\,\psi_{1,1}^{2r}(t-2l)^*\,dt = \frac{2}{N}\int_0^N \sum_{n,m=-N/4}^{N/4-1} \gamma_{1,1}^{2r}[n](t-2k)\,\gamma_{1,1}^{2r}[m](t-2l)^*\,dt$$

$$= \frac{2}{N}\sum_{n,m=-N/4}^{N/4-1} \omega^{2(kn-lm)} \int_0^N \gamma_{1,1}^{2r}[n](t)\,\gamma_{1,1}^{2r}[m](t)^*\,dt = \frac{2}{N}\sum_{n=-N/4}^{N/4-1} \omega^{2n(k-l)} = \delta[k-l].$$

Thus, $N/2$ splines $\left\{\psi_{1,1}^{2r}(t-2k)\right\}$, which are orthogonal to each other and whose norms are equal to one, form an orthonormal basis of the $N/2$-dimensional space ${}^{2r}\mathscr{W}_{1,1}$. Proof for the splines $\left\{\psi_{1,0}^{2r}(t-2k)\right\}$ is similar. ■

Figure 8.3, which was produced by the MATLAB code `Gen_wp_L1P.m`, displays the wavelets $\psi_{1,0}^{2r}(t)$ and $\psi_{1,1}^{2r}(t+1)$ for $r = 1, 2, 3, 4, 5$ and their SHA spectra

$\nu_{1,0}[n]$ and $\bar{\nu}_{1,1}[n]$, respectively. The spectra $\nu_{1,0}[n]$ occupy the central parts of the plots, which correspond to lower frequencies, while $\bar{\nu}_{1,1}[n]$ are located near the edges, which correspond to higher frequencies. The higher the spline's order is, the closer are shapes of the SHA spectra $\nu_{1,0}[n]$ and $\bar{\nu}_{1,1}[n]$ to rectangles and the smaller is their overlap. Respectively, the effective support of the wavelets widens when the spline's order increases.

8.3.3 Transforms to Coarser Levels

Assume the spline space $S(t) \in {}^{2r}\mathscr{S}$ is split into four orthogonal subspaces as in Eq. (8.20). Then a spline $S(t) \in {}^{2r}\mathscr{S}$ given in Eq. (8.28) can be decomposed into its orthogonal projections onto the subspaces ${}^{2r}\mathscr{W}_{2,l}$, $l = 0, 1, 2, 3$: $S(t) = \sum_{l=0}^{3} S_{2,l}(t)$, where

$$S_{2,l}(t) \stackrel{\text{def}}{=} \sqrt{\frac{1}{N_l}} \sum_{n=-N_2/2}^{N_2/2-1} \sigma_{2,l}[n]\gamma_{2,l}^{2r}[n](t), \quad N_2 = \frac{N}{4}.$$

Similarly to the first decomposition level, the coordinates $\sigma_{2,0}[n]$ and $\sigma_{2,1}[n]$ are derived from $\sigma_{1,0}[n]$ using the matrix $\mathbf{A}_1[n]$ defined in Eq. (8.27):

$$\begin{pmatrix} \sigma_{2,0}[n] \\ \sigma_{2,1}[n] \end{pmatrix} = \sqrt{\frac{1}{2}}\mathbf{A}_1[n] \cdot \begin{pmatrix} \sigma_{1,0}[n] \\ \sigma_{1,0}[n + N/4] \end{pmatrix}, \quad n = -N/8, \ldots, N/8 - 1.$$

The inverse relation is

$$\begin{pmatrix} \sigma_{1,0}[n] \\ \sigma_{1,0}[n + N/4] \end{pmatrix} = \sqrt{2}\mathbf{A}_1^{\dagger}[n] \cdot \begin{pmatrix} \sigma_{2,0}[n] \\ \sigma_{2,1}[n] \end{pmatrix}$$
$$= \sqrt{2} \begin{pmatrix} s_1[n] & d_1^*[n] \\ s_1[n + N/4] & d_1^*[n + N/4] \end{pmatrix} \cdot \begin{pmatrix} \sigma_{2,0}[n] \\ \sigma_{2,1}[n] \end{pmatrix}, \quad n = -N/8, \ldots, N/8 - 1.$$

Hence, the splines are embedded into the space ${}^{2r}\mathscr{W}_{1,0}$ as

$$S_{2,0}(t) = \sqrt{\frac{4}{N}} \sum_{n=-N/4}^{N/4-1} s_1[n]\,\sigma_{2,0}[n]\,\gamma_{1,0}^{2r}[n](t),$$
$$S_{2,1}(t) = \sqrt{\frac{4}{N}} \sum_{n=-N/4}^{N/4-1} d_1^*[n]\,\sigma_{2,1}[n]\,\gamma_{1,0}^{2r}[n](t).$$

Their coordinates in the space ${}^{2r}\mathscr{W}_{1,0}$ are $\sqrt{2}\,s_1[n]\,\sigma_{2,0}[n]$ and $\sqrt{2}\,d_1^*[n]\,\sigma_{2,1}[n]$, respectively. Using Eq. (8.35), we embed the splines into the initial space ${}^{2r}\mathscr{S}$, thus deriving their SHA spectra:

$$S_{2,p}(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \varsigma_{2,p}[n]\, \gamma^{2r}[n](t), \quad p = 0, 1,$$

$$\varsigma_{2,0}[n] \stackrel{\text{def}}{=} 2 s_0[n]\, s_1[n]\, \sigma_{2,0}[n] = \sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}} \sqrt{\frac{u_1^{4r}[n]}{u_2^{4r}[n]}}\, \cos^{2r} \frac{\pi n}{N}\, \cos^{2r} \frac{2\pi n}{N}\, \sigma_{2,0}[n]$$

$$= \sqrt{\frac{u^{4r}[n]}{u_2^{4r}[n]}}\, \cos^{2r} \frac{\pi n}{N}\, \cos^{2r} \frac{2\pi n}{N}\, \sigma_{2,0}[n],$$

$$\varsigma_{2,1}[n] \stackrel{\text{def}}{=} 2 s_0[n]\, d_1^*[n]\, \sigma_{2,1}[n]$$

$$= \omega^{-2n} \sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}} \sqrt{\frac{u_1^{4r}[n+N/4]}{u_2^{4r}[n]}}\, \cos^{2r} \frac{\pi n}{N}$$

$$\times \sin^{2r} \frac{2\pi n}{N}\, \sigma_{2,1}[n].$$

The sequences $\{\varsigma_{2,0}[n]\}$ and $\{\varsigma_{2,1}[n]\}$, $n = -N/2, \ldots, N/2-1$, form the SHA spectra of the splines $S_{2,0}(t) \in {}^{2r}\mathscr{W}_{2,0}$ and $S_{2,1}(t) \in {}^{2r}\mathscr{W}_{2,1}$, respectively. It is readily seen that their effective supports are $(-N/8, \ldots, N/8-1)$ and $(-N/4, \ldots, -N/8-1) \bigcup (N/8, \ldots, N/4-1)$, respectively. One should keep in mind that the sequences $\{u_1^{4r}[n]\}$, $\{\sigma_{1,0}[n]\}$ and $\{\sigma_{1,1}[n]\}$ are $N/2$-periodic, while $\{u_2^{4r}[n]\}$, $\{\sigma_{2,0}[n]\}$ and $\{\sigma_{2,1}[n]\}$ are $N/4$-periodic.

To project the spline $S(t) \in {}^{2r}\mathscr{S}$ onto the subspaces ${}^{2r}\mathscr{W}_{2,2}$ and ${}^{2r}\mathscr{W}_{2,3}$, utilize the matrix $\mathbf{A}_1[n]$ once more. For $n = -N/8, \ldots, N/8-1$, the following relations hold:

$$\begin{pmatrix} \sigma_{2,3}[n] \\ \sigma_{2,2}[n] \end{pmatrix} = \sqrt{\frac{1}{2}} \mathbf{A}_1[n] \cdot \begin{pmatrix} \sigma_{1,1}[n] \\ \sigma_{1,1}[n+N/4] \end{pmatrix},$$

$$= \sqrt{\frac{1}{2}} \begin{pmatrix} s_1[n]\sigma_{1,1}[n] + s_1[n+N/4]\sigma_{1,1}[n+N/4] \\ d_1^*[n]\sigma_{1,1}[n] + d_1^*[n+N/4]\sigma_{1,1}[n+N/4] \end{pmatrix}. \quad (8.43)$$

The inverse relation is

$$\begin{pmatrix} \sigma_{1,0}[n] \\ \sigma_{1,0}[n+N/4] \end{pmatrix} = \sqrt{2} \mathbf{A}_1^{\dagger}[n] \cdot \begin{pmatrix} \sigma_{2,3}[n] \\ \sigma_{2,2}[n] \end{pmatrix}$$

$$= \sqrt{2} \begin{pmatrix} s_1[n] & d_1^*[n] \\ s_1[n+N/4] & d_1^*[n+N/4] \end{pmatrix} \cdot \begin{pmatrix} \sigma_{2,3}[n] \\ \sigma_{2,2}[n] \end{pmatrix}.$$

Hence, the splines are embedded into the space ${}^{2r}\mathscr{W}_{1,1}$ as

$$S_{2,3}(t) = \sqrt{\frac{4}{N}} \sum_{n=-N/4}^{N/4-1} s_1[n]\, \sigma_{2,3}[n]\, \gamma_{1,1}^{2r}[n](t),$$

$$S_{2,2}(t) = \sqrt{\frac{4}{N}} \sum_{n=-N/4}^{N/4-1} d_1^*[n]\, \sigma_{2,2}[n]\, \gamma_{1,1}^{2r}[n](t).$$

Using Eq. (8.35), we embed the splines into the initial space ${}^{2r}\mathscr{S}$, thus deriving their SHA spectra:

$$\begin{aligned}
S_{2,p}(t) &= \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \varsigma_{2,p}[n]\, \gamma^{2r}[n](t), \quad p = 2, 3, \\
\varsigma_{2,3}[n] &\stackrel{\text{def}}{=} 2 d_0^*[n]\, s_1[n]\, \sigma_{2,3}[n] \\
&= \omega^{-n} \sqrt{\frac{u^{4r}[n + N/2]}{u_2^{4r}[n]}}\, \sin^{2r} \frac{\pi n}{N}\, \cos^{2r} \frac{2\pi n}{N}\, \sigma_{2,0}[n], \\
\varsigma_{2,2}[n] &\stackrel{\text{def}}{=} 2 d_0^*[n]\, d_1^*[n]\, \sigma_{2,2}[n] \\
&= \omega^{-3n} \sqrt{\frac{u^{4r}[n + N/2]}{u_1^{4r}[n]}} \sqrt{\frac{u_1^{4r}[n + N/4]}{u_2^{4r}[n]}} \times \\
&\quad \times \sin^{2r} \frac{\pi n}{N}\, \sin^{2r} \frac{2\pi n}{N}\, \sigma_{2,1}[n].
\end{aligned}$$

The sequences $\{\varsigma_{2,3}[n]\}$ and $\{\varsigma_{2,2}[n]\}$, $n = -N/2, \ldots, N/2 - 1$, form the SHA spectra of the splines $S_{2,3}(t) \in {}^{2r}\mathscr{W}_{2,3}$ and $S_{2,2}(t) \in {}^{2r}\mathscr{W}_{2,2}$, respectively.

Remark 8.3.1 At the first glance, it seems natural to derive, unlike Eq. (8.43), $\sigma_{2,2}[n]$ by application of the "low-pass p-filter" $s_1[n]$ to the sequence $\sigma_{1,1}[n]$, $n = -N/8, \ldots, N/8 - 1$, the "high-pass p-filter" $s_1[n + N/4]$ to the sequence $\sigma_{1,1}[n + N/4]$ and vice versa for $\sigma_{2,3}[n]$. However, an analysis of the structure of iterated p-filters,which is similar to the analysis in Proposition 8.7, dictates the transform in Eq. (8.43). As a result of this scheme, the effective supports of the sequences $\{\varsigma_{2,2}[n]\}$ and $\{\varsigma_{2,3}[n]\}$, $n = -N/2, \ldots, N/2 - 1$, which form the SHA spectra of the splines $S_{2,2}(t) \in {}^{2r}\mathscr{W}_{2,2}$ and $S_{2,3}(t) \in {}^{2r}\mathscr{W}_{2,3}$, are $N/4, \ldots, 3N/8 - 1$, and $3N/8, \ldots, N/2 - 1$, (and symmetrically for the negative n), respectively. Thus, the SHA spectra $\{\varsigma_{2,0-}[n]\}$, $\{\varsigma_{2,1}[n]\}$, $\{\varsigma_{2,2}[n]\}$, and $\{\varsigma_{2,3}[n]\}$ produce a split of the SHA spectrum of $S(t)$ into four parts. The indexing of these partial spectra correspond to their frequency content.

A similar interchanging scheme is implemented for the direct and inverse transforms to the subsequent level. A general rule is:

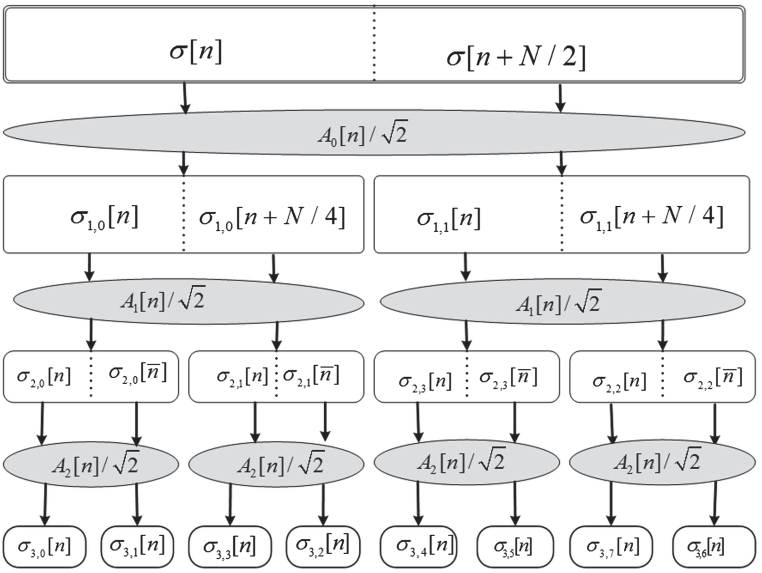


Fig. 8.4 Diagram of 3-level wavelet packet transform of a spline, whose SHA spectrum is $\{\sigma[n]\}$. Notation $\bar{n} \stackrel{\text{def}}{=} n + N/8$ for coefficients of the third level

$$\begin{pmatrix} \sigma_{m,2l}[n] \\ \sigma_{m,2l+1}[n] \end{pmatrix} = \sqrt{\frac{1}{2}}\mathbf{A}_{m-1}[n] \cdot \begin{pmatrix} \sigma_{m-1,l}[n] \\ \sigma_{m-1,1}[n+N_m] \end{pmatrix} \text{ if } l \text{ is even}, \quad (8.44)$$
$$\begin{pmatrix} \sigma_{m,2l+1}[n] \\ \sigma_{m,2l}[n] \end{pmatrix} = \sqrt{\frac{1}{2}}\mathbf{A}_{m-1}[n] \cdot \begin{pmatrix} \sigma_{m-1,l}[n] \\ \sigma_{m-1,1}[n+N_m] \end{pmatrix} \text{ if } l \text{ is odd}.$$

A diagram of the 3-level wavelet packet transform of a spline, whose SHA spectrum is $\{\sigma[n]\}$ is displayed in Fig. 8.4. For compactness, we denote $\bar{n} \stackrel{\text{def}}{=} n + N/8$ for the coefficients of the third decomposition level.

The reconstruction is implemented in an inverse order.

$$\begin{pmatrix} \sigma_{m-1,l}[n] \\ \sigma_{m-1,1}[n+N_m] \end{pmatrix} = \sqrt{2}\mathbf{A}^{\dagger}_{m-1}[n] \cdot \begin{pmatrix} \sigma_{m,2l}[n] \\ \sigma_{m,2l+1}[n] \end{pmatrix} \text{ if } l \text{ is even}, \quad (8.45)$$
$$\begin{pmatrix} \sigma_{m-1,l}[n] \\ \sigma_{m-1,1}[n+N_m] \end{pmatrix} = \sqrt{2}\mathbf{A}^{\dagger}_{m-1}[n] \cdot \begin{pmatrix} \sigma_{m,2l+1}[n] \\ \sigma_{m,2l}[n] \end{pmatrix} \text{ if } l \text{ is odd}.$$

Remark 8.3.2 In order to derive the SHA spectrum $\{\varsigma_{m,l}[n]\}$, $n = -N/2, \ldots, N/2-1$, of a spline

$$S(t) = \frac{1}{\sqrt{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \sigma_{m,l}[n]\, \gamma^{2r}_{m,l}[n](t) \in {}^{2r}\mathscr{W}_{m,l},$$

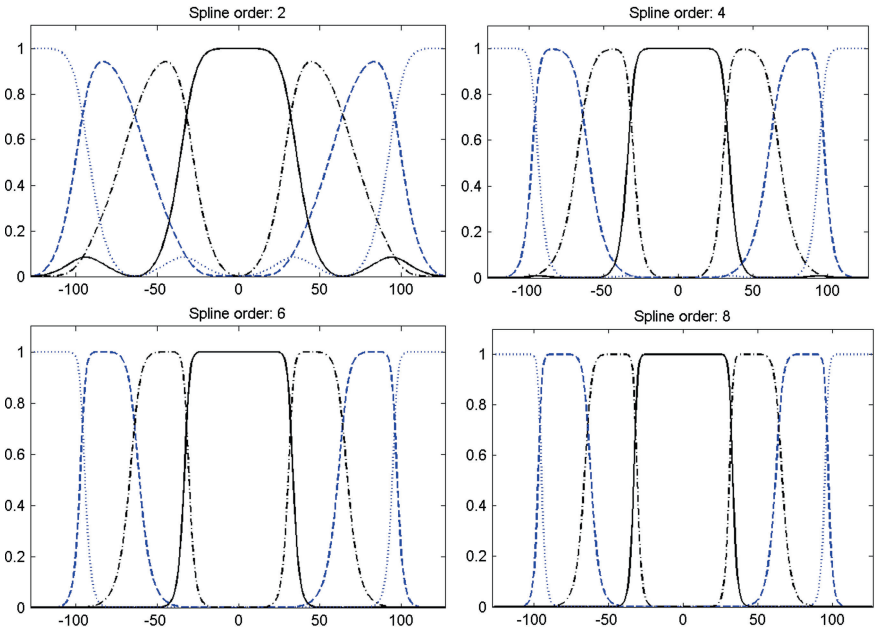


Fig. 8.5 The magnitude responses of the second level s-filters $s_{2,0}[n]$ (*solid curves*), $s_{2,1}[n]$ (*dashdot curves*), $s_{2,2}[n]$ (*dashed curves*) and $s_{2,3}[n]$ (*dotted curves*) for the splines of orders 2, 4, 6 and 8, $N = 256$

we put $\sigma_{m,\tilde{l}}[n] = 0$ for all $n = -N_m/2, \ldots, N_m/2 - 1$ and $\tilde{l} \neq l$. Then, the reconstruction operations (8.45) should be iteratively implemented starting from the level m till the initial level.

Denote the frequency responses of the second level p-filters , which generate the SHA spectra of the orthogonal projections of a spline, as

$$s_{2,0}[n] \stackrel{\text{def}}{=} s_0[n]\, s_1[n],\ \ s_{2,1}[n] \stackrel{\text{def}}{=} s_0[n]\, d_1^*[n],$$
$$s_{2,2}[n] \stackrel{\text{def}}{=} d_0^*[n]\, d_1^*[n],\ \ s_{2,3}[n] \stackrel{\text{def}}{=} d_0^*[n]\, d_1^*[n],$$

where $n = -N/2, \ldots, N/2 - 1$. The magnitude responses of the p-filters are displayed in Fig. 8.5, which was produced by the MATLAB code `S_filters.m`.

Remark 8.3.3 Note that a spline $S_{2,p}(t) \in {}^{2r}\mathscr{W}_{2,p}$, $p = 0, 1, 2, 3$, from the second decomposition level

$$S_{2,p}(t) = \sqrt{\frac{1}{N_2}} \sum_{n=-N_2/2}^{N_2/2-1} \sigma_{2,p}[n]\, \gamma_{2,p}^{2r}[n](t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \varsigma_{2,p}[n]\, \gamma^{2r}[n](t),$$

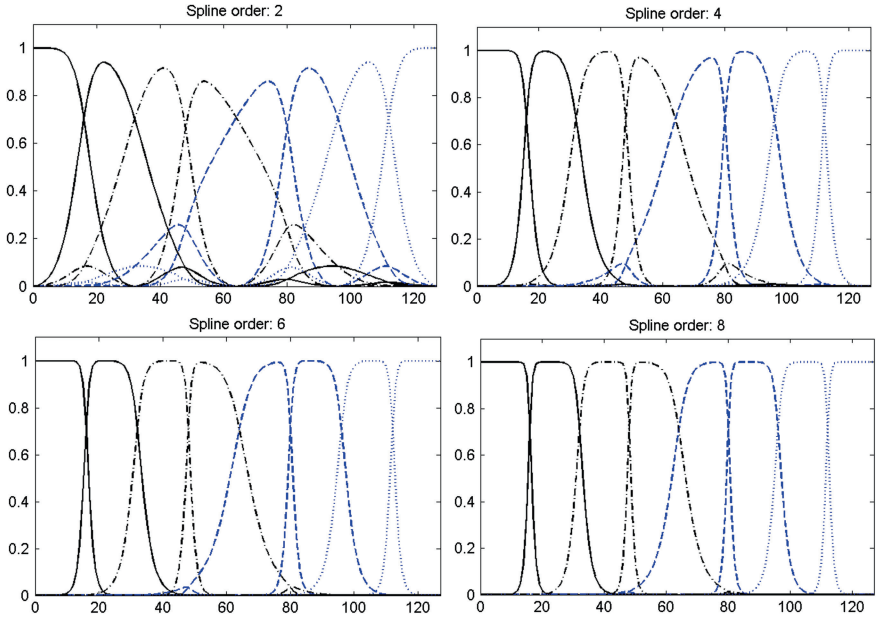


Fig. 8.6 The magnitude responses of the third level p-filters for the splines of orders 2, 4, 6 and 8, $N = 256$

where $\varsigma_{2,p}[n] = s_{2,p}[n]\,\sigma_{2,p}[n]$ can be regarded as a result of s-filtering the spline

$$\tilde{S}_{2,p}(t) = N^{-1/2} \sum_{n=-N/2}^{N/2-1} \sigma_{2,p}[n]\,\gamma^{2r}[n](t) \in {}^{2r}\mathscr{S}$$

with the s-filter whose frequency response is $s_{2,p}[n]$. Recall that the sequences $\sigma_{2,p}[n]$ are $N_2 = N/4$-periodic. A similar observation is true for coarser decomposition levels.

Figure 8.6 displays the magnitude responses of the third level s-filters. For better resolution of the picture, in Fig. 8.6 the magnitude responses are displayed for the half-band $n = 0, \ldots, N/2-1$. The other half-band $n = -N/2, \ldots, -1$ mirrors the former one. We observe that these magnitude responses provide refined partitions of a spline' SHA spectrum into 4 or 8 components, respectively, whose overlap is minimal. The shapes of the curves tend to rectangles as the splines' orders increase.

The plots in Figs. 8.5 and 8.6 were produced by the MATLAB code `S_filters.m`.

8.4 Spline Wavelet Packets

8.4.1 Definition of Spline Wavelet Packets

The complex-valued basis splines $\gamma_m^{2r}[n](t)$ and $\gamma_{m,l}^{2r}[n](t)$, which are designed iteratively via two-scale relations (8.21) and (8.22), are well localized in the frequency domain, but their supports in the time domain occupy the whole interval $[0, N)$ (up to periodization). Extending the definition of the periodic Battle-Lemarié wavelets in Eqs. (8.39) and (8.40), we introduce a family of orthonormal bases for the spline space ${}^{2r}\mathscr{S}$. Elements of these family are real-valued and well localized in time domain. Denote,

$$\psi_{m,l}^{2r}(t) \stackrel{\text{def}}{=} \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \gamma_{m,l}^{2r}[n](t) \in {}^{2r}\mathscr{W}_{m,l}, \quad l = 0, \dots 2^m - 1. \quad (8.46)$$

Proposition 8.10 *There holds the relation reciprocal to Eq.* (8.46)*:*

$$\gamma_{m,l}^{2r}[n](t) = \sqrt{\frac{1}{N_m}} \sum_{k=0}^{N_m-1} \omega^{2^m kn}\, \psi_{m,l}^{2r}\left(t - 2^m k\right). \quad (8.47)$$

Proof Equation (8.46) and the shift property Eq. (8.23) imply that

$$\psi_{m,l}^{2r}\left(t - 2^m k\right) = \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{-2^m kn}\, \gamma_{m,l}^{2r}[n](t).$$

Applying the IDFT, we get Eq. (8.47). ■

Proposition 8.11 *The translations* $\left\{\psi_{m,l}^{2r}(t - 2^m k)\right\}, \quad k = 0, \dots, N_m - 1,$*, where* $N_m = N/2^m$*, form an orthonormal basis for the space* ${}^{2r}\mathscr{W}_{m,l}$*. The translations*

$$\bigcup_{l=0}^{2^m-1} \left\{\psi_{m,l}^{2r}\left(t - 2^m k\right)\right\}, \quad k = 0, \dots, N_m - 1,$$

form an orthonormal basis for the entire space ${}^{2r}\mathscr{S}$*.*

Proof The spline $\psi_{m,l}^{2r}(t - 2^m k)$ is orthogonal to any spline $\psi_{m,\tilde{l}}^{2r}\left(t - 2^m \tilde{k}\right)$, for $\tilde{l} \neq l$ because they belong to mutually orthogonal subspaces. The inner product of two splines from the same subspace is

$$\int_0^N \psi_{m,l}^{2r}\left(t-2^m k\right)\,\psi_{m,l}^{2r}\left(t-2^m\tilde{k}\right)^*\,dt$$
$$=\frac{1}{N_m}\sum_{n,\tilde{n}=-N_m/2}^{N_m/2-1}\omega^{-2^m(kn-\tilde{k}\tilde{n})}\int_0^N \gamma_{m,l}^{2r}[n](t)\gamma_{m,l}^{2r}[\tilde{n}]^*(t)\,dt$$
$$=\frac{1}{N_m}\sum_{n=-N_m/2}^{N_m/2-1} w^{-2^m n(k-\tilde{k})}=\delta[k-\tilde{k}],\quad k,\,\tilde{k}=0,\ldots,N_m-1.$$

Thus, N_m mutually orthogonal splines $\left\{\psi_{m,l}^{2r}\left(t-2^m k\right)\right\}$, $k=0,\ldots,N_m-1$, whose norms are equal to one, form a basis of the N_m-dimensional space ${}^{2r}\mathscr{W}_{m,l}$. ■

The splines $\psi_{m,0}^{2r}(t)$ and $\psi_{m,1}^{2r}(t)$ are periodic Battle-Lemarié father and mother wavelets, respectively. The splines $\psi_{m,l}^{2r}(t)$ with arbitrary $l=1,\ldots,2^m-1$, are periodic orthonormal wavelet packets.

8.4.2 The SHA Spectra of Spline Wavelet Packets

All the spaces ${}^{2r}\mathscr{W}_{m,l}$ are the subspaces of the initial spline space ${}^{2r}\mathscr{S}$, thus the wavelet packets $\psi_{m,l}^{2r}(t)$ belong to ${}^{2r}\mathscr{S}$. To efficiently operate with them, we need to know the SHA spectra $\left\{\nu_{m,l}[n]\right\}_{n=-N/2}^{N/2-1}$ of these wavelet packets, which are their coordinates in the orthonormal basis $\left\{\gamma^{2r}[n](t)\right\}$, $n=-N/2,\ldots,N/2-1$, of the space ${}^{2r}\mathscr{S}$:

$$\psi_{m,l}^{2r}(t)\stackrel{\text{def}}{=}\sqrt{\frac{1}{N_m}}\sum_{n=-N_m/2}^{N_m/2-1}\gamma_{m,l}^{2r}[n](t)=\sqrt{\frac{1}{N}}\sum_{n=-N/2}^{N/2-1}\nu_{m,l}[n]\gamma^{2r}[n](t).\quad(8.48)$$

At the initial scale we have $\varphi^{2r}(t)=\psi_0^{2r}(t)=N^{-1/2}\sum_{n=-N/2}^{N/2-1}\gamma^{2r}[n](t)\Longrightarrow \nu_0[n]=1$. The SHA spectra of the first level wavelet packets $\psi_{1,0}^{2r}(t)$ and $\psi_{1,1}^{2r}(t)$ were presented in Eq. (8.41)

$$\nu_{1,0}[n]=\sqrt{2}s_0[n]=\sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N},$$
$$\nu_{1,1}[n]=\sqrt{2}d_0^*[n]=\omega^{-n}\sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N}.$$

Generally, in line with Remark 8.3.2, to derive the SHA spectrum $\left\{\nu_{m,l}[n]\right\}$, $n=-N/2,\ldots,N/2-1$, of a wavelet packet $\psi_{m,l}^{2r}(t)$, one should put $\sigma_{m,l}[n]=1$,

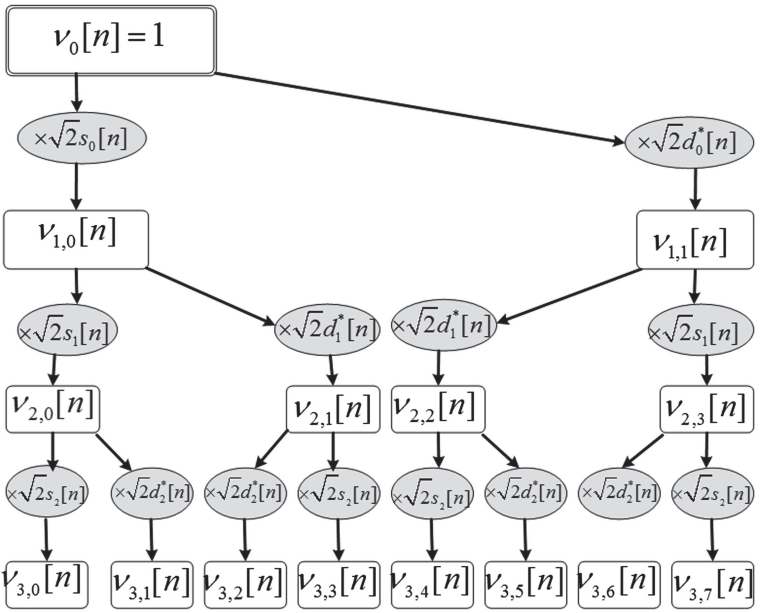


Fig. 8.7 Diagram of calculation of the SHA spectra of 3-level wavelet packets

$\sigma_m[n]=0$ and $\sigma_{m,\tilde{l}}[n]=0$ for all $n=-N_m/2,\ldots,N_m/2-1$ and $\tilde{l}\neq l$. Then, the reconstruction operations (8.45) should be iteratively implemented, starting from the level m till the initial level. However, practically, the calculations can be implemented by the following tree-structured scheme.

$$
\begin{aligned}
&\nu_0[n]=1,\quad \begin{pmatrix}\nu_{1,0}[n]\\ \nu_{1,1}[n]\end{pmatrix}=\sqrt{2}\,\nu_0[n]\begin{pmatrix}s_0[n]\\ d_0^*[n]\end{pmatrix},\\
&\ldots\ldots\ldots\ldots\\
&\begin{pmatrix}\nu_{m,2l}[n]\\ \nu_{m,2l+1}[n]\end{pmatrix}=\sqrt{2}\,\nu_{m-1,l}[n]\begin{pmatrix}s_{m-1}[n]\\ d_{m-1}^*[n]\end{pmatrix}\text{ if } l \text{ is even},\\
&\begin{pmatrix}\nu_{m,2l}[n]\\ \nu_{m,2l+1}[n]\end{pmatrix}=\sqrt{2}\,\nu_{m-1,l}[n]\begin{pmatrix}d_{m-1}^*[n]\\ s_{m-1}[n]\end{pmatrix}\text{ if } l \text{ is odd},\quad n=-N/2,\ldots N/2-1.
\end{aligned}
\tag{8.49}
$$

The scheme is illustrated by the diagram in Fig. 8.7.

Remark 8.4.1 It is seen from Eqs. (8.49) and the diagram in Fig. 8.7 that the SHA spectra of the pair of m-level wavelet packets $\psi_{m,2l}^{2r}(t)$ and $\psi_{m,2l+1}^{2r}(t)$ are derived from the SHA spectrum of the $m-1$-level wavelet packet $\psi_{m,l}^{2r}(t)$. In that sense, the wavelet packet $\psi_{m,l}^{2r}(t)$ is the "parent" of $\psi_{m,2l}^{2r}(t)$ and $\psi_{m,2l}^{2r}(t)$ while $\psi_{m,2l+1}^{2r}(t)$ and $\psi_{m,2l+1}^{2r}(t)$ are its "offsprings".

Example: Wavelet packets from the second level

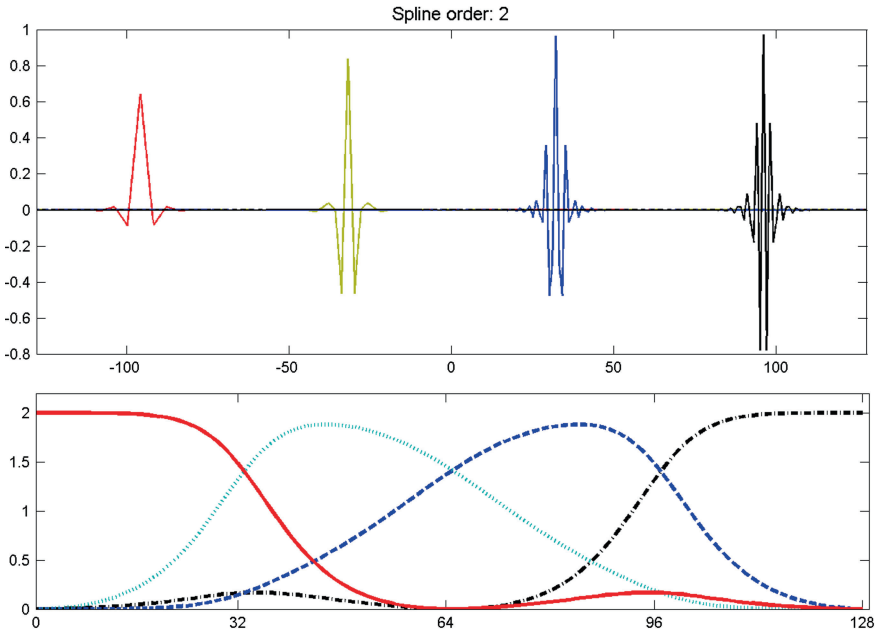


Fig. 8.8 Second level wavelet packets and the right half-band of their SHA spectra. Upper frame from *left* to *right*: $\psi^2_{2,0}(t)$, $\psi^2_{2,1}(t)$, $\psi^2_{2,2}(t)$, $\psi^2_{2,3}(t)$. Bottom frame: $\nu_{2,0}[n]$, (*solid line*) $\nu_{2,1}[n]$ (*dotted*), $\nu_{2,2}[n]$ (*dashed*), $\nu_{2,3}[n]$ (*dashdotted*)

$$\nu_{2,0}[n] = 2s_0[n]\,s_1[n] = \sqrt{\frac{u^{4r}[n]}{u_2^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N}\,\cos^{2r}\frac{2\pi n}{N},$$

$$\nu_{2,1}[n] = 2s_0[n]\,d_1^*[n] = \omega^{-2n}\sqrt{\frac{u^{4r}[n]}{u_1^{4r}[n]}}\sqrt{\frac{u_1^{4r}[n+N/4]}{u_2^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N}\,\sin^{2r}\frac{2\pi n}{N},$$

$$\nu_{2,2}[n] = 2d_0^*[n]\,d_1^*[n] = \omega^{-3n}\sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\sqrt{\frac{u_1^{4r}[n+N/4]}{u_2^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N}\,\sin^{2r}\frac{2\pi n}{N},$$

$$\nu_{2,3}[n] = 2d_0^*[n]\,d_1^*[n] = \omega^{-n}\sqrt{\frac{u^{4r}[n+N/2]}{u_1^{4r}[n]}}\sqrt{\frac{u_1^{4r}[n]}{u_2^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N}\,\cos^{2r}\frac{2\pi n}{N},$$

where $n = -N/2, , N/2 - 1$. Figures 8.8, 8.9 and 8.10 display the second level wavelet packets and their SHA spectra for the splines of orders 2, 4 and 8, respectively. These figures were prepared by the MATLAB code `wp_level2P.m`. The SHA spectra of the wavelet packets and their grid samples were produced by the function `wavs_coefP.m`.

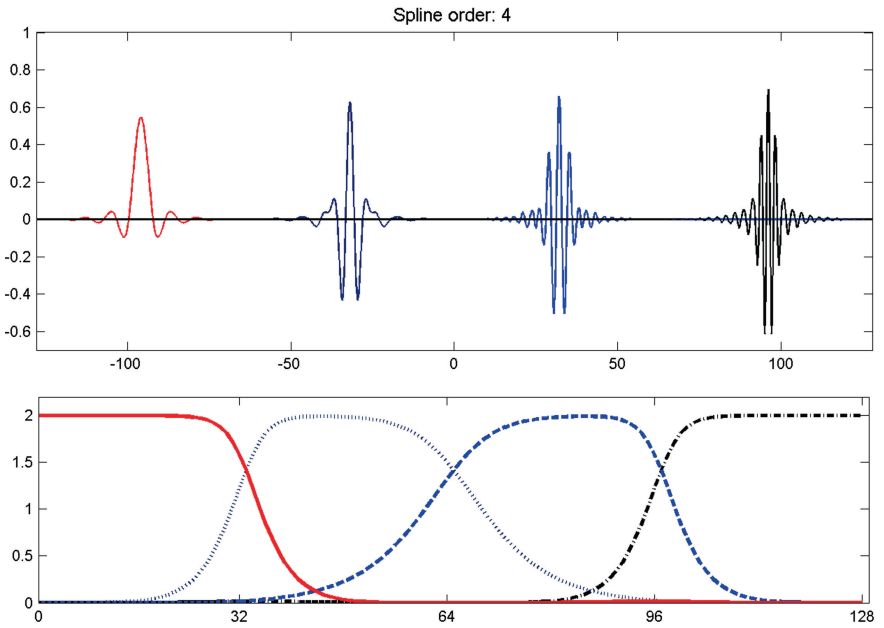


Fig. 8.9 Second level wavelet packets and right half-band of their SHA spectra. Upper frame from *left* to *right*: $\psi^4_{2,0}(t)$, $\psi^4_{2,1}(t)$, $\psi^4_{2,2}(t)$, $\psi^4_{2,3}(t)$. Bottom frame: $v_{2,0}[n]$, (*solid line*) $v_{2,1}[n]$ (*dotted*), $v_{2,2}[n]$ (*dashed*), $v_{2,3}[n]$ (*dashdotted*)

8.4.3 Wavelet Packets Bases

Assume that a spline $S(t)$ belongs to the space ${}^{2r}\mathscr{W}_{m,l}$. Then, it can be expanded over the orthonormal wavelet packets basis. Equation (8.46) and the shift property (8.19) imply that

$$\begin{aligned} S(t) &= \sum_{k=0}^{N_m-1} q_{m,l}[k]\, \psi^{2r}_{m,l}\left(t-2^m k\right) = \sqrt{\frac{1}{N_m}} \sum_{k=0}^{N_m-1} q_{m,l}[k] \sum_{n=-N_m/2}^{N_m/2-1} \gamma^{2r}_{m,l}[n]\left(t-2^m k\right) \\ &= \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \gamma^{2r}_{m,l}[n]\,(t) \sum_{k=0}^{N_m-1} \omega^{-2^m nk}\, q_{m,l}[k] \\ &= \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \sigma_{m,l}[n]\, \gamma^{2r}_{m,l}[n]\,(t)\,. \end{aligned}$$

Thus, the coordinates of the spline $S(t) \in {}^{2r}\mathscr{W}_{m,l}$ in two orthonormal bases of the space are linked via the DFT as

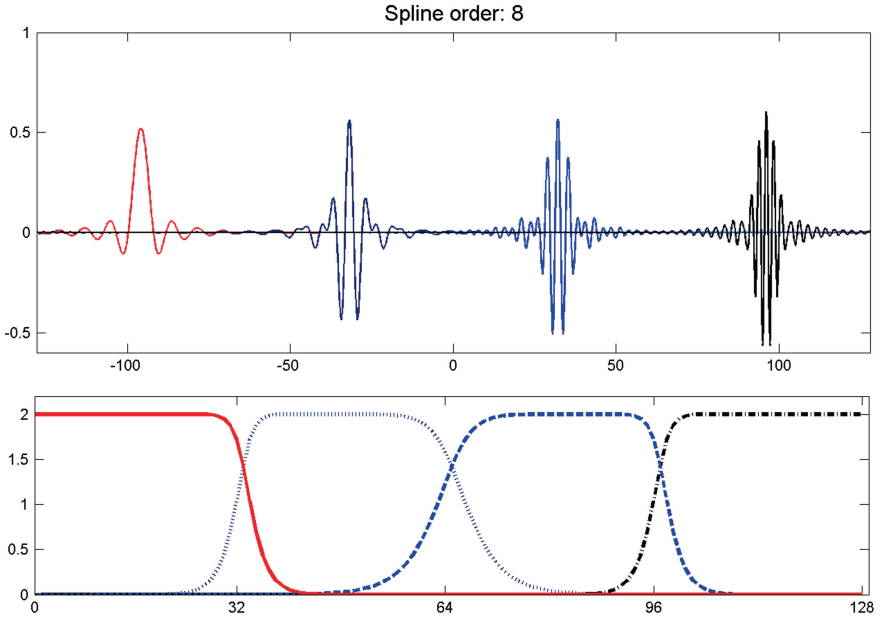


Fig. 8.10 Second level wavelet packets and right half-band of their SHA spectra. Upper frame from *left* to *right*: $\psi^8_{2,0}(t)$, $\psi^8_{2,1}(t)$, $\psi^8_{2,2}(t)$, $\psi^8_{2,3}(t)$. Bottom frame: $v_{2,0}[n]$, (*solid line*) $v_{2,1}[n]$ (*dotted*), $v_{2,2}[n]$ (*dashed*), $v_{2,3}[n]$ (*dashdotted*)

$$\sigma_{m,l}[n] = \sum_{k=0}^{N_m-1} \omega^{-2^m nk}\, q_{m,l}[k], \quad q_{m,l}[k] = \frac{1}{N_m} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{2^m nk}\, \sigma_{m,l}[n]. \quad (8.50)$$

In particular, if a spline $S(t)$ belongs to the initial space ${}^{2r}\mathscr{S}$, then it can be expanded over the alternative orthonormal bases:

$$S(t) = \sum_{k=0}^{N-1} q[k]\varphi^{2r}(t-k) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\gamma^{2r}[n](t),$$

where $\varphi^{2r}(t)$ is the periodic Battle–Lemarié father wavelet defined in Eq. (8.38). The coordinates are linked via the DFT

$$\sigma[n] = \sum_{k=0}^{N-1} \omega^{-nk}\, q[k], \quad q[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \sigma[n]. \quad (8.51)$$

Assume, the set $\{\sigma[n]\}$, $n = -N/2, \ldots, N/2-1$ is available. For example, if we know the grid values of the spline: $S(k) = z[k]$, $k = 0, \ldots, N-1$ then, due to Eq. (8.52),

$$\sigma[n] = \frac{\hat{z}[n]}{V^{2r}[n]}, \quad V^{2r}[n] = \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}, \quad \hat{z}[n] = \sum_{k=0}^{N-1} \omega^{-nk} z[k]. \tag{8.52}$$

Consequently, all the coordinates $\sigma_{m,l}[n]$ of the orthogonal projections $S(t)$ of the spline $S(t)$ onto the subspaces ${}^{2r}\mathscr{W}_{m,l}, m = 1, \ldots, M,\ l = 0, \ldots, 2^m - 1$ are calculated by arithmetic operations (Eq. (8.44) and Fig. 8.4). Then, by utilizing Eq. (8.50), we find the coordinates $q_{m,l}[k]$ in the wavelet packets' bases $\left\{\psi_{m,l}^{2r}\left(t - 2^m k\right)\right\}$, respectively, of the projections $S(t)$. These computations are fast, because they consist of forward and backward application of the fast Fourier transform. The wavelet packet transform of a spline, whose grid samples are known, is implemented by the MATLAB function `samp_spl_WP_analP.m`. Once these coordinates are calculated, it is possible to represent the spline $S(t)$ via a variety of orthonormal bases, which are constituted by the wavelet packets $\left\{\psi_{m,l}^{2r}\left(t - 2^m k\right)\right\}$ that belong to different combinations of the subspaces ${}^{2r}\mathscr{W}_{m,l}$. The obvious options are:

- "Horizontal" bases that are bases consisting of shifts of all the wavelet packets from the same level m, where m can be one of the integers $1, \ldots, M$:

$$\mathbf{B}_m^h \stackrel{\text{def}}{=} \bigcup_{l=0}^{L} \left\{\psi_{m,l}^{2r}\left(t - 2^m k\right)\right\}, \tag{8.53}$$

  where $L = 2^m - 1$ and $k = 0, \ldots, N_m - 1$, $N_m = N/2^m$. In this case, the SHA spectrum of a spline $S(t)$ is split into L regions of the same size as seen, for example, in Figs. 8.8, 8.9 and 8.10. Restoration of a row signal or of rows of a 2D array from its coordinates in a "horizontal" wavelet packets' basis is implemented by the MATLAB function `horizontal_WP_synthP.m`.
- Wavelet bases

$$\mathbf{B}_m^w \stackrel{\text{def}}{=} \left\{\psi_{m,0}^{2r}\left(t - 2^m k\right)\right\}_{k=0}^{N_m-1} \bigcup \left\{\psi_{m,1}^{2r}\left(t - 2^m k\right)\right\}_{k=0}^{N_m-1} \tag{8.54}$$
$$\bigcup \left\{\psi_{m-1,1}^{2r}\left(t - 2^{m-1} k\right)\right\}_{k=0}^{N_{m-1}-1} \bigcup \ldots \bigcup \left\{\psi_{1,1}^{2r}\left(t - 2k\right)\right\}_{k=0}^{N/2-1}.$$

  The SHA spectrum of a spline $S(t)$ is split in a logarithmic mode. Figures 8.11, 8.12 and 8.13 display wavelets $\psi_{3,0}^{2r}(t)$, $\psi_{3,1}^{2r}(t)$, $\psi_{2,1}^{2r}(t)$, $\psi_{1,1}^{2r}(t)$ and their SHA spectra, which originate from the splines of order 2, 4 and 8, respectively. The plots were produced by the MATLAB code `wW_level3P.m`.

The direct and inverse wavelet transforms of the spline coordinates are implemented by the MATLAB functions `spl_Wv_analP` and `spl_Wv_syntP`, respectively.

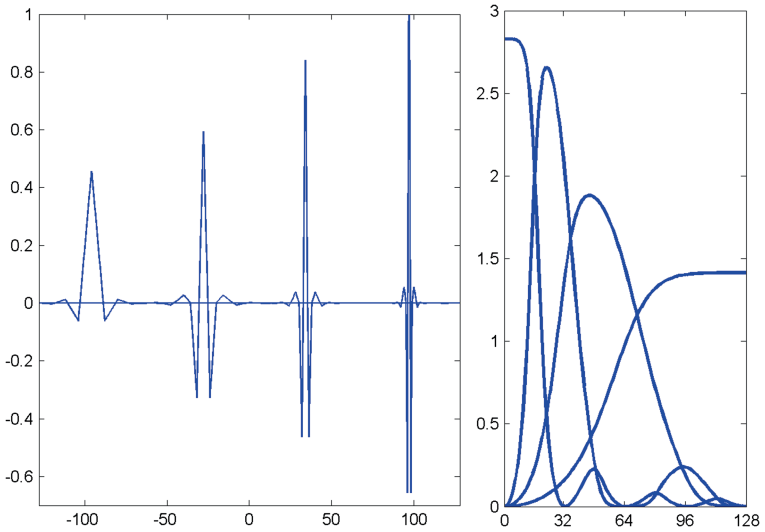


Fig. 8.11 Spline wavelets of order 2 and *right* half-band of their SHA spectra. *Left* frame from *left* to *right*: $\psi^2_{3,0}(t)$, $\psi^2_{3,1}(t)$, $\psi^2_{2,1}(t)$, $\psi^2_{1,1}(t)$. *Right* frame: $\nu_{3,0}[n]$, $\nu_{3,1}[n]$, $\nu_{2,1}[n]$, $\nu_{1,1}[n]$

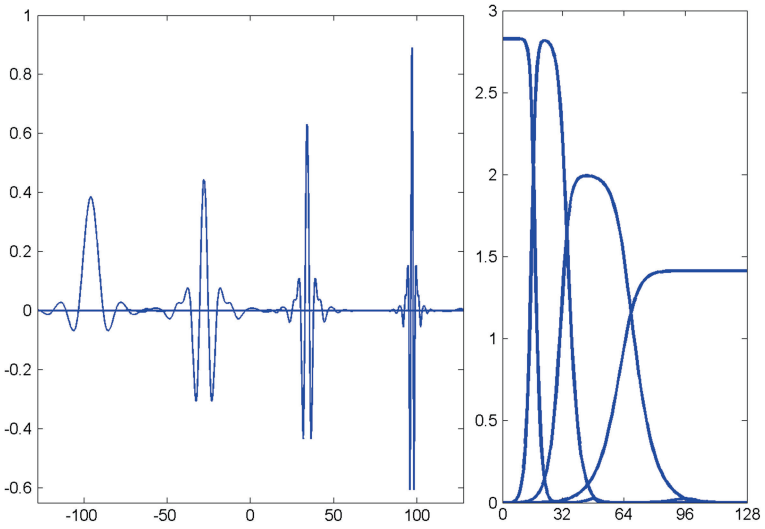


Fig. 8.12 Spline wavelets of order 4 and *right* half-band of their SHA spectra. *Left* frame from *left* to *right*: $\psi^4_{3,0}(t)$, $\psi^4_{3,1}(t)$, $\psi^4_{2,1}(t)$, $\psi^4_{1,1}(t)$. *Right* frame: $\nu_{3,0}[n]$, $\nu_{3,1}[n]$, $\nu_{2,1}[n]$, $\nu_{1,1}[n]$

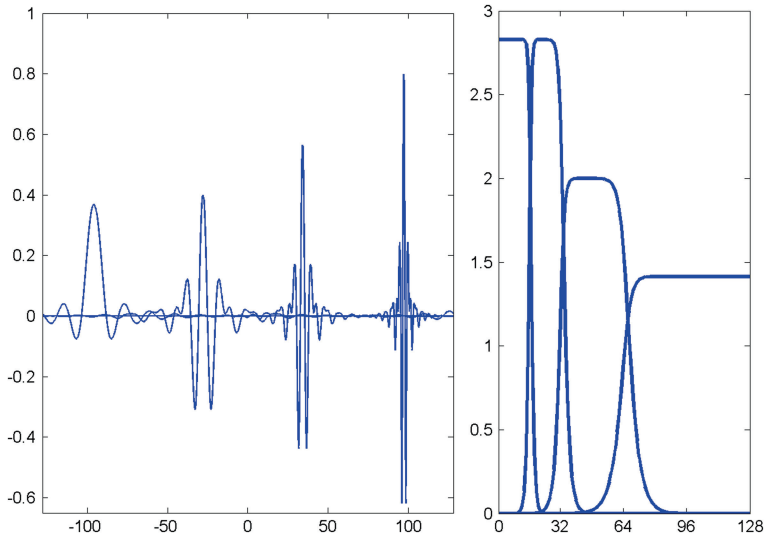


Fig. 8.13 Spline wavelets of order 8 and *right* half-band of their SHA spectra. *Left* frame from *left* to *right*: $\psi^8_{3,0}(t)$, $\psi^8_{3,1}(t)$, $\psi^8_{2,1}(t)$, $\psi^8_{1,1}(t)$. *Right* frame: $\nu_{3,0}[n]$, $\nu_{3,1}[n]$, $\nu_{2,1}[n]$, $\nu_{1,1}[n]$

8.4.4 Best Basis

Besides two kinds of orthonormal wavelet packets bases, which are presented above, a variety of orthonormal wavelet packets bases is possible. These bases are generated by shifts of wavelet packets from various decomposition levels. A combination of wavelet packets can generate a basis for the spline space ${}^{2r}\mathscr{S}$ if their SHA spectra form a one-fold cover of the whole spectral domain $n = -N/2, \ldots N/2 - 1$.

Remark 8.4.2 We stress that an orthonormal basis of a space ${}^{2r}\mathscr{W}_{m-1,l}$ can be constituted either by the set of shifts $\left\{\psi^{2r}_{m-1,l}\left(t - 2^{m-1}k\right)\right\}$, $k = 0, \ldots, N_{m-1} - 1$, of the wavelet packet $\psi^{2r}_{m-1,l}(t)$ or by the union

$$\left\{\psi^{2r}_{m,2l}\left(t - 2^m k\right)\right\} \bigcup \left\{\psi^{2r}_{m,2l+1}\left(t - 2^m k\right)\right\}, \; k = 0, \ldots, N_m - 1,$$

of the sets of shifts of its "offsprings" $\psi^{2r}_{m,2l}(t)$ and $\psi^{2r}_{m,2l+1}(t)$.

Examples:
The following plots of the SHA spectra illustrate different configurations of the wavelet packets' bases.

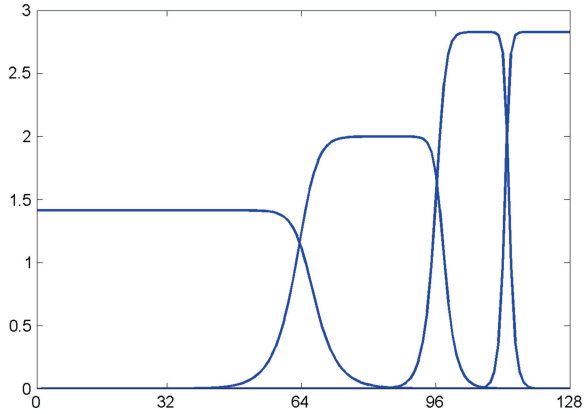


Fig. 8.14 SHA spectra of the spline wavelet packets of eighth order: $\psi_{1,0}^{8}(t)$, $\psi_{2,2}^{8}(t)$, $\psi_{3,6}^{8}(t)$ and $\psi_{3,7}^{8}(t)$

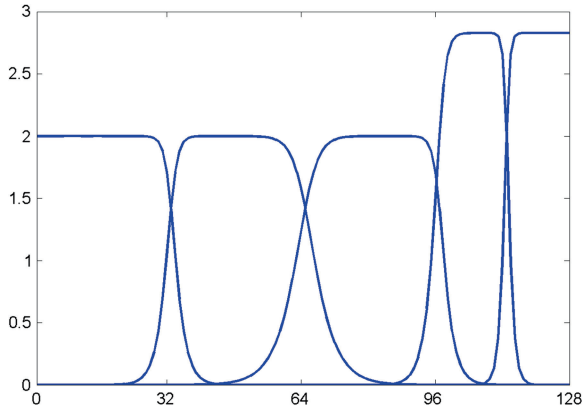


Fig. 8.15 SHA spectra of the spline wavelet packets of eighth order: $\psi_{2,0}^{8}(t)$, $\psi_{2,1}^{8}(t)$, $\psi_{2,2}^{8}(t)$, $\psi_{3,6}^{8}(t)$ and $\psi_{3,7}^{8}(t)$

1.

$$\mathbf{B}_3^1 \stackrel{\text{def}}{=} \left\{\psi_{1,0}^{2r}(t-2k)\right\}_{k=0}^{N/2-1} \bigcup \left\{\psi_{2,2}^{2r}(t-4k)\right\}_{k=0}^{N/4-1} \bigcup \left\{\psi_{3,6}^{2r}(t-8k)\right\}_{k=0}^{N/8-1} \bigcup \left\{\psi_{3,7}^{2r}(t-8k)\right\}_{k=0}^{N/8-1}.$$

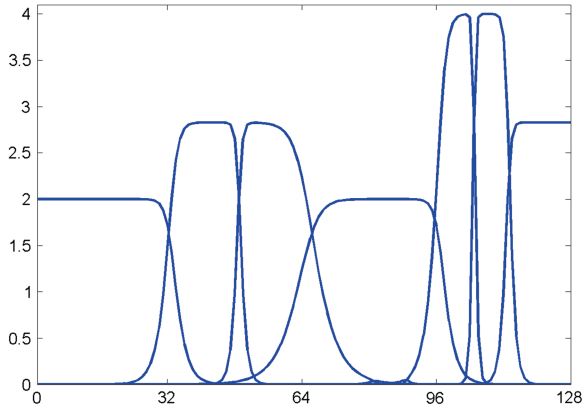


Fig. 8.16 SHA spectra of the spline wavelet packets of order 8: $\psi^8_{2,0}(t)$, $\psi^8_{3,2}(t)$,$\psi^8_{3,3}(t)$, $\psi^8_{2,2}(t)$, $\psi^8_{4,12}(t)$, $\psi^8_{4,13}(t)$ and $\psi^8_{3,7}(t)$

2. The wavelet $\psi^{2r}_{1,0}(t)$ can be replaced by the pair of its "offsprings":

$$\mathbf{B}_3^2 \stackrel{\text{def}}{=} \left\{\psi^{2r}_{2,0}(t-4k)\right\}_{k=0}^{N/4-1} \bigcup \left\{\psi^{2r}_{2,2}(t-4k)\right\}_{k=0}^{N/4-1} \bigcup \left\{\psi^{2r}_{1,2}(t-4k)\right\}_{k=0}^{N/4-1}$$
$$\bigcup \left\{\psi^{2r}_{3,6}(t-8k)\right\}_{k=0}^{N/8-1} \bigcup \left\{\psi^{2r}_{3,7}(t-8k)\right\}_{k=0}^{N/8-1}.$$

3. The wavelet packets $\psi^{2r}_{2,1}(t)$, and $\psi^{2r}_{3,6}(t)$ can be replaced by their"offsprings":

$$\mathbf{B}_4^3 \stackrel{\text{def}}{=} \left\{\psi^{2r}_{2,0}(t-4k)\right\}_{k=0}^{N/4-1} \bigcup \left\{\psi^{2r}_{3,2}(t-8k)\right\}_{k=0}^{N/8-1} \bigcup \left\{\psi^{2r}_{3,3}(t-8k)\right\}_{k=0}^{N/8-1}$$
$$\bigcup \left\{\psi^{2r}_{2,2}(t-4k)\right\}_{k=0}^{N/4-1} \bigcup \left\{\psi^{2r}_{4,12}(t-16k)\right\}_{k=0}^{N/16-1}$$
$$\bigcup \left\{\psi^{2r}_{4,13}(t-16k)\right\}_{k=0}^{N/16-1} \bigcup \left\{\psi^{2r}_{3,7}(t-8k)\right\}_{k=0}^{N/8-1}.$$

Figures 8.14, 8.15 and 8.16 were produced by the MATLAB code `shasp_plotP`. In order to select an orthonormal basis, which provides an optimal representation of the spline $S(t)$ for a certain problem, one should start with a definition of a cost function, which enables to compare the feasibility of different bases to the given problem.

Definition 8.1 The best basis for a spline with respect to a chosen cost function is the basis, which minimizes the cost function.

In many problems, such as compression, denoising,deconvolution, to name a few, a representation, which concentrates the spline's energy in a relatively small number of big coefficients, the remaining coefficients being negligibly small, is regarded as optimal. For that representation, a good choice of the cost function for a given orthonormal basis **B** is the Shannon entropy [11] of spline's coordinates

$\mathbf{q} \stackrel{\text{def}}{=} \{q[k]\}\,,\ k = 0, \ldots, N-1$, in this basis

$$E(\mathbf{B}) \stackrel{\text{def}}{=} -\sum_{k=0}^{N-1} \frac{|q[k]|^2}{\|\mathbf{q}\|^2}\ \log_e \frac{|q[k]|^2}{\|\mathbf{q}\|^2} \tag{8.55}$$

The entropy measures the flatness of the energy distribution of the spline. Minimizing the entropy leads to an efficient representation of the spline when its energy is concentrated in a few essential waveforms.

For different types of problems, other cost functions are used. For example, in classifications problems the so-called discriminant measures are utilized [1, 9, 10].

Generally, calculation and comparison of costs of all possible orthonormal bases is a computationally expensive task, especially in multidimensional cases. However, when the cost function is additive, the Best Basis algorithm by Coifman and Wickerhauser [4, 14] significantly reduces the amount of computations.

Definition 8.2 Assume, an orthonormal basis $\mathbf{B}$ is the union of two bases $\mathbf{B} = \mathbf{B}^1 \bigcup \mathbf{B}^2$. The cost function C is additive if for any such basis $C(\mathbf{B}) = C(\mathbf{B}^1) + C(\mathbf{B}^2)$.

The entropy is an example of the additive cost function.

Selection of the best wavelet packet orthonormal basis for a spline $S(t) \in {}^{2r}\mathscr{S}$ is based on the following proposition, which determines the best bases of the spline's projections.

Proposition 8.12 [4] *Let $\mathbf{B}_{m-1,l}$ be a wavelet packet orthonormal basis for the projection of a spline $S(t) \in {}^{2r}\mathscr{S}$ onto the space ${}^{2r}\mathscr{W}_{m-1,l}$ and $\bar{\mathbf{B}}_{m,2l}$ and $\bar{\mathbf{B}}_{m,2l+1}$ be the best bases for the subspaces ${}^{2r}\mathscr{W}_{m,2l}$ and ${}^{2r}\mathscr{W}_{m,2l+1}$, respectively. If C is an additive cost function, then the best basis for the space ${}^{2r}\mathscr{W}_{m-1,l}$ is*

$$\bar{\mathbf{B}}_{m-1,l} = \begin{cases} \bar{\mathbf{B}}_{m,2l} \bigcup \bar{\mathbf{B}}_{m,2l+1}, & \text{if } C\left(\mathbf{B}_{m-1,l}\right) < C\left(\bar{\mathbf{B}}_{m,2l}\right) + C\left(\bar{\mathbf{B}}_{m,2l+1}\right); \\ \mathbf{B}_{m-1,l}, & \text{if } C\left(\mathbf{B}_{m-1,l}\right) \geq C\left(\bar{\mathbf{B}}_{m,2l}\right) + C\left(\bar{\mathbf{B}}_{m,2l+1}\right). \end{cases} \tag{8.56}$$

Design of the best wavelet packet basis for a spline $S(t) \in {}^{2r}\mathscr{S}$ consists of the following steps:

1. Implement wavelet packet transform of the spline $S(t)$ down to a level M.
2. Calculate the cost function $C_{m,l} \stackrel{\text{def}}{=} C(\left\{\psi_{m,l}^{2r}\left(t - 2^m k\right)\right\})$ for all the bases of the subspaces ${}^{2r}\mathscr{W}_{m,l}$, $m = 0, \ldots M$, of the space ${}^{2r}\mathscr{S} = {}^{2r}\mathscr{W}_{0,0}$.
3. The best (and the single available) basis $\bar{\mathbf{B}}^M$ for the bottom level M is the union of the wavelet packet bases of the subspaces ${}^{2r}\mathscr{W}_{M,l}$, $l = 0, \ldots, 2^M - 1$.
4. The best basis $\bar{\mathbf{B}}^{M-1}$ for the upper level $M-1$ is designed in line with the rule Eq. (8.56) by comparing the cost functions of the "parent" subspaces with ${}^{2r}\mathscr{W}_{M-1,l}$ with the cost functions of their "offsprings" ${}^{2r}\mathscr{W}_{M,2l}$ and ${}^{2r}\mathscr{W}_{M,2l+1}$.
5. The procedure is iterated up to the top space ${}^{2r}\mathscr{W}_{0,0}$.

Once the wavelet packet transform of a signal is implemented and the entropies in each block of the transform coefficients are calculated by the MATLAB function `spl_WP_analP.m`, the list of the wavelet packets, whose shifts constitute the best basis tree is compiled by the function `WP_BB_Listree1D_P.m`.

8.4.5 Reconstruction of a Spline from Wavelet Packet Bases

Assume a spline $S(t) \in {}^{2r}\mathscr{S}$ is expanded over some wavelet packet basis. The basis is constituted by shifts $\left\{\psi_{m,l}^{2r}\left(t-2^m k\right)\right\}$ of a set of orthonormal wavelet packets $\left\{\psi_{m,l}^{2r}(t)\right\}$, $(m,l) \subset \Lambda$. That means that the spline is represented as the orthogonal sum

$$S(t) = \sum_{(m,l)\subset\Lambda} S_{m,l}(t), \quad S_{m,l}(t) = \sum_{k=0}^{N/2^m-1} q_{m,l}[k]\,\psi_{m,l}^{2r}\left(t-2^m k\right).$$

The wavelet packets $\psi_{m,l}^{2r}(t)$ can be expanded over orthonormal basis of exponential splines as $\psi_{m,l}^{2r}(t) = N^{-1/2}\sum_{n=-N/2}^{N/2-1} \nu_{m,l}[n]\,\gamma^{2r}[n](t)$, where the set $\nu_{m,l}[n]$ is the SHA spectrum of $\psi_{m,l}^{2r}(t)$. Then, using the shift property Eq. (4.20) of the exponential splines, we have

$$\begin{aligned} S_{m,l}(t) &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N/2^m-1} q_{m,l}[k] \sum_{n=-N/2}^{N/2-1} \nu_{m,l}[n]\,\gamma^{2r}[n]\left(t-2^m k\right) \\ &= \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \gamma^{2r}[n](t)\,\nu_{m,l}[n] \sum_{k=0}^{N/2^m-1} \omega^{-2^m kn}\, q_{m,l}[k] \\ &= \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma_{m,l}[n]\,\gamma^{2r}[n](t), \quad \sigma_{m,l}[n] \stackrel{\text{def}}{=} \nu_{m,l}[n]\,\hat{q}_{m,l}[n]_m. \end{aligned} \tag{8.57}$$

Consequently, the spline $S(t)$ is restored as

$$S(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\,\gamma^{2r}[n](t) = \sum_{k=0}^{N-1} q[k]\,\varphi^{2r}(t-k),$$

$$\sigma[n] = \sum_{(m,l)\subset\Lambda} \sigma_{m,l}[n], \quad q[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\,\sigma[n]$$

where $\sigma_{m,l}[n]$ are defined in Eq. (8.57). The grid samples of the splines are

$$S(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sigma[n] \, \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} \, \omega^{kn}.$$

The MATLAB function `List_synthP_samp.m` restores the spline $S(t)$ from the basis defined by a list Λ. It produces the coefficients of the spline $\{q[k]\}$ in the orthonormal basis $\left\{\varphi^{2r}(t-k)\right\}$, the SHA spectrum of the spline and its grid samples.

8.5 2D Wavelet Packets

Assume that the 2D splines $S(x, y)$ defined on the grid $\{k, n\}$ are $N-$periodic and have the order $p = 2r$ in both x and y directions. The space of such splines is denoted as ${}^{2r,2r}\mathscr{S}$. These splines were described in Sect. 4.2.

8.5.1 2D Wavelet Packets Bases

The splines $S(x, y) \in {}^{2r,2r}\mathscr{S}$ can be expanded over the orthonormal basis consisting of 2D normalized exponential splines $\gamma^{2r}[\kappa, \iota](x, y) \stackrel{\text{def}}{=} \gamma^{2r}[\kappa](x) \cdot \gamma^{2r}[\iota](y)$:

$$S(x, y) = \frac{1}{N} \sum_{\kappa,\iota=-N/2}^{N/2-1} \sigma[\kappa, \iota] \;\; \gamma^{2r}[\kappa, \iota](x, y). \tag{8.58}$$

If the grid samples of the spline $S(x, y) \in {}^{2r,2r}\mathscr{S}$ are known: $S(k, n) = z[k, n]$, k, $n = 0, \ldots, N-1$, then, Eq. (4.62) provides coordinates of the spline:

$$\sigma[\kappa, \nu] = \frac{\hat{z}[\kappa, \nu]}{V^{2r}[\kappa] \, V^{2r}[\nu]}, \quad V^{2r}[n] \stackrel{\text{def}}{=} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}, \quad \hat{z}[\kappa, \nu] = \sum_{\kappa,\nu=0}^{N-1} \omega^{-(\kappa k + \nu n)} \, z[k, n].$$

Denote

$$\gamma^{2r}_{m,l,\tilde{l}}[\kappa, \iota](x, y) \stackrel{\text{def}}{=} \gamma^{2r}_{m,l}[\kappa](x) \; \gamma^{2r}_{m,\tilde{l}}[\iota](y), \quad l, \tilde{l} = 0, \ldots, 2^m - 1.$$

Obviously, the splines $\gamma^{2r}_{m,l,\tilde{l}}[\kappa, \iota](x, y)$ with the same level index m are mutually orthogonal and their norms

$$\left\| \gamma^{2r}_{m,l,\tilde{l}}[\kappa, \iota] \right\| = \sqrt{\int_0^N \left| \gamma^{2r}_{m,l,\tilde{l}}[\kappa, \iota](x, y) \right|^2 \, dx \, dy} = 1.$$

The 2D spline spaces ${}^{2r}\mathscr{W}_{m,l,\tilde{l}} \subset {}^{2r,2r}\mathscr{S}$ are defined as the linear spans of the splines $\gamma^{2r}_{m,l,\tilde{l}}[\kappa,\iota](x,y),\ l,\tilde{l}=0,\dots 2^m-1$, respectively, where $N_m = N/2^m$:

$$S_{m,l,\tilde{l}}(x,y) = \frac{1}{N_m}\sum_{\kappa,\iota=-N_m/2}^{N_m/2-1} \sigma_{m,l,\tilde{l}}[\kappa,\iota]\gamma^{2r}_{m,l,\tilde{l}}[\kappa,\iota](x,y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}. \quad (8.59)$$

The splits (8.24) of the 1D spline spaces generate the splits of the space ${}^{2r,2r}\mathscr{S}$ into mutually orthogonal subspaces:

$${}^{2r}\mathscr{W}_{m-1,l,\tilde{l}} = {}^{2r}\mathscr{W}_{m,2l,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{m,2l,2\tilde{l}+1} \bigoplus {}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}+1}. \quad (8.60)$$

Hence, different representations of the initial space follow

$$\begin{aligned} {}^{2r,2r}\mathscr{S} &= {}^{2r,2r}\mathscr{W}_{1,0,0} \bigoplus {}^{2r}\mathscr{W}_{1,0,1} \bigoplus {}^{2r}\mathscr{W}_{1,1,0} \bigoplus {}^{2r}\mathscr{W}_{1,1,1} \\ &\quad \cdots\cdots\cdots \\ &= \bigoplus_{l,\tilde{l}=1}^{2^m-1} {}^{2r}\mathscr{W}_{m,l,\tilde{l}}, \quad m=2,\dots,M. \end{aligned}$$

As an alternative to the orthonormal basis $\left\{\gamma^{2r}_{m,l,\tilde{l}}[\kappa,\iota](x,y)\right\},\ \kappa,\iota=-N_m/2,\dots,N_m/2-1$, of a subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}} \subset {}^{2r,2r}\mathscr{S}$, there exists an orthonormal basis that consist of shifts of the 2D wavelet packets. Then, the spline expansion , which is alternative to Eq. (8.59), is

$$S_{m,l,\tilde{l}}(x,y) = \sum_{k,n=0}^{N_m-1} q_{m,l,\tilde{l}}[k,n]\psi^{2r}_{m,l,\tilde{l}}(x-2^m k, y-2^m n) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}, \quad (8.61)$$

where

$$\begin{aligned} \psi^{2r}_{m,l,\tilde{l}}(x,y) &\stackrel{\text{def}}{=} \psi^{2r}_{m,l}(x)\,\psi^{2r}_{m,\tilde{l}}(y) = \frac{1}{N_m}\sum_{\kappa,\iota=-N_m/2}^{N_m/2-1} \gamma^{2r}_{m,l}[\kappa](x)\,\gamma^{2r}_{m,\tilde{l}}[\iota](y), \\ q_{m,l,\tilde{l}}[k,n] &= \frac{1}{N_m}\sum_{\kappa,\iota=-N_m/2}^{N_m/2-1} \omega^{2^m(n\iota+k\kappa)}\,\sigma_{m,l,\tilde{l}}[\kappa,\iota]. \end{aligned} \quad (8.62)$$

The spline $\psi^{2r}_{m,0,0}(x,y)$ is the 2D periodic father wavelet from level m, and the triple $\psi^{2r}_{m,1,0}(x,y)$, $\psi^{2r}_{m,0,1}(x,y)$ and $\psi^{2r}_{m,1,1}(x,y)$ are the 2D periodic wavelets from level m.

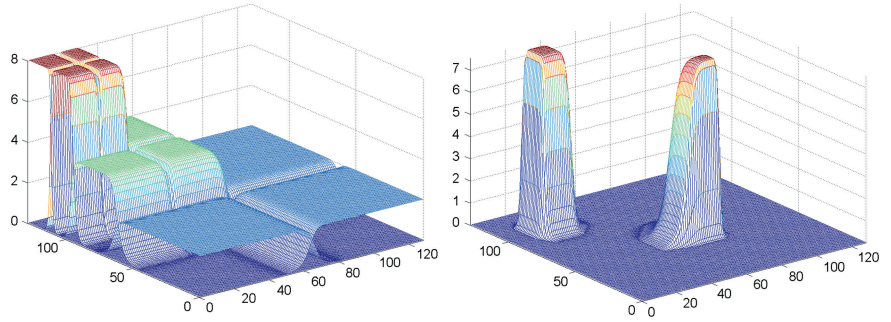


Fig. 8.17 *Left* The SHA spectra of wavelets of order 10 down to the third decomposition level. *Right* SHA spectra of the wavelet packets of order 10 from the third decomposition level: $\psi^{10}_{3,2,2}(x, y)$ and $\psi^{10}_{3,5,5}(x, y)$

Two-dimensional wavelets and wavelet packets can be expanded over the orthonormal basis $\left\{\gamma^{2r}[\kappa, \iota](x, y) = \gamma^{2r}[\kappa](x)\gamma^{2r}[\iota](y)\right\}$ of the initial space ${}^{2r,2r}\mathscr{S}$:

$$\psi^{2r}_{m,l,\tilde{l}}(x, y) = \frac{1}{N} \sum_{\kappa,\iota=-N/2}^{N/2-1} v_{m,l,\tilde{l}}[\kappa, \iota]\, \gamma^{2r}[\kappa, \iota](x, y),$$

and their SHA spectra $\left\{v_{m,l,\tilde{l}}[\kappa, \iota] = v_{m,l}[\kappa]v_{m,\tilde{l}}[\iota]\right\}$ are the tensor products of the SHA spectra of the 1D wavelet packets. Figure 8.17 displays the SHA spectra of the 2D wavelets of order 10 down to the third decomposition level and of two wavelet packets of order 10 from the third scale. For clarity, we display only the first quadrant $\kappa, \iota = 0, \ldots, N/2 - 1$ of the spectral square. The magnitudes of SHA spectra in the other quadrants mirror those in first quadrant.One can observe that the spectra have near–parallelepiped shape. The figure was prepared by the MATLAB code `mesh_wp_listP.m`.

8.5.2 2D Wavelet Packets Transforms

Assume a spline $S(x, y) \in {}^{2r}\mathscr{S}$ is represented as in Eq. (8.58) and $\{\sigma[\kappa, \iota]\}$ constitute its SHA spectrum. Then, the coordinates $\left\{\sigma_{m,l,\tilde{l}}[\kappa, \iota]\right\}$, $m = 1, \ldots, M$, of its projections $S_{m,l,\tilde{l}}(x, y)$ (Eq. (8.59)) onto the subspaces ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$, respectively, are derived by application of the 2D wavelet packet transforms to the coordinates $\{\sigma[\kappa, \iota]\}, \kappa, \iota = -N/2, \ldots, N/2 - 1$. The 2D transform can be implemented by iterated application of the 1D transform subsequently to the rows and columns of the array $\left\{\sigma_{m,0,0}[\kappa, \iota]\right\}$.

Assume a spline $S_{m,l,\tilde{l}}(x, y) \in {}^{2r}\mathscr{S}_{m,l,\tilde{l}}$ is expanded as in Eq. (8.59). The space ${}^{2r}\mathscr{S}_{m,l,\tilde{l}}$ is a subspace of the initial spline space ${}^{2r,2r}\mathscr{S}$. Thus, the spline $S_{m,l,\tilde{l}}(x, y)$ can be expanded over the orthonormal basis $\left\{\gamma^{2r}[\kappa, \iota]\right\}$, $\kappa, \iota = -N/2, \ldots, N/2-1$, of the space ${}^{2r,2r}\mathscr{S}$:

$$S_{m,l,\tilde{l}}(x, y) = \frac{1}{N} \sum_{\kappa,\iota=-N/2}^{N/2-1} \varsigma_{m,l,\tilde{l}}[\kappa, \iota]\, \gamma^{2r}[\kappa, \iota](x, y). \tag{8.63}$$

The set of coordinated $\left\{\varsigma_{m,1,\tilde{1}}[\kappa, \iota]\right\}$, $\kappa, \iota = -N/2, \ldots, N/2-1$, constitutes its SHA spectrum. The SHA spectrum can be derived by application of the 2D inverse wavelet packet transforms to the coefficients array $\left\{\sigma_{m,l,\tilde{l}}[\kappa, \iota]\right\}$.

8.5.2.1 Matrix Expression of One-Level 2D Wavelet Packet Transforms

Similarly to 1D transforms, the direct and inverse 2D wavelet packet transforms can be explicitly expressed in a matrix form. To derive the one-level direct transform of a spline $S(x, y)$ presented in Eq. (8.58), we use relations (8.30) and (8.31). For brevity, denote $\bar{\kappa} \stackrel{\text{def}}{=} \kappa + N/2$ and $\bar{\iota} \stackrel{\text{def}}{=} \iota + N/2$. Applying these relations to the columns of the array $\{\sigma[\kappa, \iota]\}$, we get for any $\iota = -N/2, \ldots, N/2-1$ and $\kappa = -N/4, \ldots, N/4-1$

$$\sigma^c_{1,0}[\kappa, \iota] = \sqrt{\frac{1}{2}}\, \left(s_0[\kappa]\, \sigma[\kappa, \iota] + s_0[\bar{\kappa}]\, \sigma[\bar{\kappa}, \iota]\right),$$

$$\sigma^c_{1,1}[\kappa, \iota] = \sqrt{\frac{1}{2}}\, \left(d_0[\kappa]\, \sigma[\kappa, \iota] + d_0[\bar{\kappa}]\, \sigma[\bar{\kappa}, \iota]\right),$$

$$\sqrt{\frac{u^{4r}[n]}{2\, u_1^{4r}[n]}}\, \cos^{2r} \frac{\pi n}{N},\ d_0[n] = \omega^n \sqrt{\frac{u^{4r}[n + N/2]}{2\, u_1^{4r}[n]}}\, \sin^{2r} \frac{\pi n}{N} = \omega^n\, s_0[n + N/2].$$

Then, relations (8.30) and (8.31) are applied to the rows of the produced arrays. For any $\kappa = -N/4, \ldots, N/4-1$ and $\iota = -N/4, \ldots, N/4-1$, we get

$$\begin{aligned} \sigma_{1,0,0}[\kappa, \iota] &= \sqrt{\frac{1}{2}}\, \left(s_0[\iota]\, \sigma^c_{1,0}[\kappa, \iota] + s_0[\bar{\iota}]\, \sigma^c_{1,0}[\kappa, \bar{\iota}]\right) \qquad (8.64) \\ &= \frac{1}{2}\, \left(s_0[\kappa]\, s_0[\iota] \sigma[\kappa, \iota] + s_0[\kappa]\, s_0[\bar{\iota}] \sigma[\kappa, \bar{\iota}] + s_0[\bar{\kappa}]\, s_0[\iota] \sigma[\bar{\kappa}, \iota]\right. \\ &\quad \left. + s_0[\bar{\kappa}]\, s_0[\bar{\iota}] \sigma[\bar{\kappa}, \bar{\iota}]\right), \\ \sigma_{1,1,0}[\kappa, \iota] &= \sqrt{\frac{1}{2}}\, \left(d_0[\iota]\, \sigma^c_{1,0}[\kappa, \iota] + d_0[\bar{\iota}]\, \sigma^c_{1,0}[\kappa, \bar{\iota}]\right) \qquad (8.65) \\ &= \frac{1}{2}\, \left(s_0[\kappa]\, d_0[\iota] \sigma[\kappa, \iota] + s_0[\kappa]\, d_0[\bar{\iota}] \sigma[\kappa, \bar{\iota}] + s_0[\bar{\kappa}]\, d_0[\iota] \sigma[\bar{\kappa}, \iota]\right. \\ &\quad \left. + s_0[\bar{\kappa}]\, d_0[\bar{\iota}] \sigma[\bar{\kappa}, \bar{\iota}]\right), \end{aligned}$$

$$\begin{aligned}\sigma_{1,0,1}[\kappa,\iota] &= \sqrt{\frac{1}{2}}\left(s_0[\iota]\,\sigma^c_{1,1}[\kappa,\iota] + s_0[\bar{\iota}]\,\sigma^c_{1,1}[\kappa,\bar{\iota}]\right) \qquad (8.66)\\ &= \frac{1}{2}\,(d_0[\kappa]\,s_0[\iota]\sigma[\kappa,\iota] + d_0[\kappa]\,s_0[\bar{\iota}]\sigma[\kappa,\bar{\iota}] + d_0[\bar{\kappa}]\,s_0[\iota]\sigma[\bar{\kappa},\iota]\\ &\quad + d_0[\bar{\kappa}]\,s_0[\bar{\iota}]\sigma[\bar{\kappa},\bar{\iota}])\,,\\ \sigma_{1,1,1}[\kappa,\iota] &= \sqrt{\frac{1}{2}}\left(d_0[\iota]\,\sigma^c_{1,1}[\kappa,\iota] + d_0[\bar{\iota}]\,\sigma^c_{1,1}[\kappa,\bar{\iota}]\right) \qquad (8.67)\\ &= \frac{1}{2}\,(d_0[\kappa]\,d_0[\iota]\sigma[\kappa,\iota] + d_0[\kappa]\,d_0[\bar{\iota}]\sigma[\kappa,\bar{\iota}] + d_0[\bar{\kappa}]\,d_0[\iota]\sigma[\bar{\kappa},\iota]\\ &\quad + d_0[\bar{\kappa}]\,d_0[\bar{\iota}]\sigma[\bar{\kappa},\bar{\iota}])\,.\end{aligned}$$

Equations (8.64)–(8.67) can be presented in a matrix form:

$$\begin{pmatrix}\sigma_{1,0,0}[\kappa,\iota]\\ \sigma_{1,1,0}[\kappa,\iota]\\ \sigma_{1,0,1}[\kappa,\iota]\\ \sigma_{1,1,1}[\kappa,\iota]\end{pmatrix} = \frac{1}{2}\mathbf{A}_0[\kappa,\iota]\cdot\begin{pmatrix}\sigma[\kappa,\iota]\\ \sigma[\kappa,\bar{\iota}]\\ \sigma[\bar{\kappa},\iota]\\ \sigma[\bar{\kappa},\bar{\iota}]\end{pmatrix}, \quad \kappa,\iota = -N/4,\ldots,N/4-1, \quad (8.68)$$

where

$$\mathbf{A}_0[\kappa,\iota] \stackrel{\text{def}}{=} \begin{pmatrix} s_0[\kappa]\,s_0[\iota] & s_0[\kappa]\,s_0[\bar{\iota}] & s_0[\bar{\kappa}]\,s_0[\iota] & s_0[\bar{\kappa}]\,s_0[\bar{\iota}]\\ s_0[\kappa]\,d_0[\iota] & s_0[\kappa]\,d_0[\bar{\iota}] & s_0[\bar{\kappa}]\,d_0[\iota] & s_0[\bar{\kappa}]\,d_0[\bar{\iota}]\\ d_0[\kappa]\,s_0[\iota] & d_0[\kappa]\,s_0[\bar{\iota}] & d_0[\bar{\kappa}]\,s_0[\iota] & d_0[\bar{\kappa}]\,s_0[\bar{\iota}]\\ d_0[\kappa]\,d_0[\iota] & d_0[\kappa]\,d_0[\bar{\iota}] & d_0[\bar{\kappa}]\,d_0[\iota] & d_0[\bar{\kappa}]\,d_0[\bar{\iota}]\end{pmatrix}$$

Comparing the matrix $\mathbf{A}_0[\kappa,\iota]$ with the matrix $\mathbf{A}_0[n]$ defined in Eq. (8.27), we observe that it can be expressed in a block-wise form:

$$\mathbf{A}_0[\kappa,\iota] \stackrel{\text{def}}{=} \begin{pmatrix} s_0[\kappa]\,\mathbf{A}_0[\iota] & s_0[\bar{\kappa}]\,\mathbf{A}_0[\iota]\\ d_0[\kappa]\,\mathbf{A}_0[\iota] & d_0[\bar{\kappa}]\mathbf{A}_0[\iota]\end{pmatrix} = \mathbf{A}_0[\kappa]\bigotimes\mathbf{A}_0[\iota],$$

where $\mathbf{A}_0[\kappa]\bigotimes\mathbf{A}_0[\iota]$ is the tensor or Kronecker product of the matrices $\mathbf{A}_0[\kappa]$ and $\mathbf{A}_0[\iota]$ [5].

The operations of conjugate transpose and inversion of the Kronecker product matrix are applied separately to its product terms. Thus, we have

$$\begin{aligned}&\mathbf{A}_0^{-1}[\kappa,\iota] = \mathbf{A}_0^{-1}[\kappa]\bigotimes\mathbf{A}_0^{-1}[\iota] = \mathbf{A}_0^{\dagger}[\kappa]\bigotimes\mathbf{A}_0^{\dagger}[\iota] = \mathbf{A}_0^{\dagger}[\kappa,\iota]\\ &= \begin{pmatrix} s_0[\kappa]\,\mathbf{A}_0^{\dagger}[\iota] & d_0^*[\kappa]\,\mathbf{A}_0^{\dagger}[\iota]\\ s_0[\bar{\kappa}]\,\mathbf{A}_0^{\dagger}[\iota] & d_0^*[\bar{\kappa}]\,\mathbf{A}_0^{\dagger}[\iota]\end{pmatrix}\\ &= \begin{pmatrix} s_0[\kappa]\,s_0[\iota] & s_0[\kappa]\,d_0^*[\iota] & d_0^*[\kappa]\,s_0[\iota] & d_0^*[\kappa]\,d_0^*[\iota]\\ s_0[\kappa]\,s_0[\bar{\iota}] & s_0[\kappa]\,d_0^*[\bar{\iota}] & d_0^*[\kappa]\,s_0[\bar{\iota}] & d_0^*[\kappa]\,d_0^*[\bar{\iota}]\\ s_0[\bar{\kappa}]\,s_0[\iota] & s_0[\bar{\kappa}]\,d_0^*[\iota] & d_0^*[\bar{\kappa}]\,s_0[\iota] & d_0^*[\bar{\kappa}]\,d_0^*[\iota]\\ s_0[\bar{\kappa}]\,s_0[\bar{\iota}] & s_0[\bar{\kappa}]\,d_0^*[\bar{\iota}] & d_0^*[\bar{\kappa}]\,s_0[\bar{\iota}] & d_0^*[\bar{\kappa}]\,d_0^*[\bar{\iota}]\end{pmatrix}.\end{aligned}$$

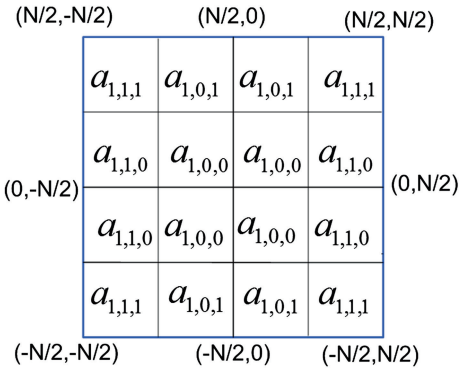


Fig. 8.18 A scheme of passbands of the 2D p-filters $\mathbf{a}_{1,0,0}$, $\mathbf{a}_{1,1,0}$, $\mathbf{a}_{1,0,1}$ and $\mathbf{a}_{1,1,1}$

Consequently, the inverse transform is presented as

$$\begin{pmatrix} \sigma[\kappa,\iota] \\ \sigma[\kappa,\bar{\iota}] \\ \sigma[\bar{\kappa},\iota] \\ \sigma[\bar{\kappa},\bar{\iota}] \end{pmatrix} = 2\mathbf{A}_0^{\dagger}[\kappa,\iota]\cdot\begin{pmatrix} \sigma_{1,0,0}[\kappa,\iota] \\ \sigma_{1,1,0}[\kappa,\iota] \\ \sigma_{1,0,1}[\kappa,\iota] \\ \sigma_{1,1,1}[\kappa,\iota] \end{pmatrix}, \quad \kappa,\iota = -N/4,\ldots,N/4-1, \tag{8.69}$$

where $\bar{\kappa} = \kappa + N/2$, $\bar{\iota} = \iota + N/2$. Hence, it follows

$$\begin{aligned}\sigma[\kappa,\iota] &= s_0[\kappa]\,s_0[\iota]\,\sigma_{1,0,0}[\kappa,\iota] + s_0[\kappa]\,d_0^*[\iota]\,\sigma_{1,1,0}[\kappa,\iota] \\ &\quad + d_0^*[\kappa]\,s_0[\iota]\,\sigma_{1,0,1}[\kappa,\iota] + d_0^*[\kappa]\,d_0^*[\iota]\,\sigma_{1,1,1}[\kappa,\iota] \\ &= s_0[\kappa]\,s_0[\iota]\,\sigma_{1,0,0}[\kappa,\iota] + \omega^{-\iota}\,s_0[\kappa]\,s_0[\iota+N/2]\,\sigma_{1,1,0}[\kappa,\iota] \\ &\quad + \omega^{-\kappa}s_0[\kappa+N/2]\,s_0[\iota]\,\sigma_{1,0,1}[\kappa,\iota] + \omega^{-\kappa-\iota}s_0[\kappa+N/2]\,s_0[\iota+N/2] \\ &\quad \times \sigma_{1,1,1}[\kappa,\iota],\end{aligned}$$

In particular, for the father wavelet $\psi^{2r}_{1,0,0}(x,y)$ the coefficients $\{\sigma_{1,0,0}[\kappa,\iota]\} = 1$, while all the other coefficients arrays $\{\sigma_{1,1,0}[\kappa,\iota]\}$, $\{\sigma_{1,0,1}[\kappa,\iota]\}$ and $\{\sigma_{1,1,1}[\kappa,\iota]\}$ are zero. Thus, its SHA spectrum is $\{s_0[\kappa]\,s_0[\iota]\}$. Similarly, the SHA spectra of the wavelets $\psi^{2r}_{1,1,0}(x,y)$, $\psi^{2r}_{1,0,1}(x,y)$ and $\psi^{2r}_{1,1,1}(x,y)$ are $\omega^{-\iota}\,s_0[\kappa]\,s_0[\iota+N/2]$, $\omega^{-\kappa}s_0[\kappa+N/2]\,s_0[\iota]$, and $\omega^{-\kappa-\iota}s_0[\kappa+N/2]\,s_0[\iota+N/2]$, respectively.

Remark 8.5.1 Keeping in mind properties of the sequences $s_0[n]$ and $d_0[n]$, which are described in Proposition 8.7, we may claim that multiplication of the matrix $\mathbf{A}_0[\kappa,\iota]$ with the vector $(\sigma[\kappa,\iota],\sigma[\kappa,\bar{\iota}],\sigma[\bar{\kappa},\iota],\sigma[\bar{\kappa},\bar{\iota}])^T$ (Eq. (8.68)) acts as application of an anti-aliasing 2D filter bank to the array of the coordinates of a spline $S(x,y)$ in the orthonormal exponential splines basis.

Denote the rows of the matrix from top to bottom $\mathbf{A}_0[\kappa,\iota]$ as $\mathbf{a}_{1,0,0}$, $\mathbf{a}_{1,1,0}$, $\mathbf{a}_{1,0,1}$, $\mathbf{a}_{1,1,1}$, respectively. They can be regarded as 2D band-pass p-filters. The passbands of these p-filters are schematically displayed in Fig. 8.18.

Thus, Eq. (8.68) means that the coefficients arrays $\{\sigma_{1,0,0}[\kappa,\iota]\}$, $\{\sigma_{1,1,0}[\kappa,\iota]\}$, $\{\sigma_{1,0,1}[\kappa,\iota]\}$ and $\{\sigma_{1,1,1}[\kappa,\iota]\}$ are derived mainly from the components $\{\sigma[\kappa,\iota]\}$,

$\{\sigma[\kappa,\bar{\iota}]\}$, $\{\sigma[\bar{\kappa},\iota]\}$, $\{\sigma[\bar{\kappa},\bar{\iota}]\}$ of the SHA spectrum of $S(x,y)$, respectively. As for the reconstruction of $S(x,y)$ from the orthogonal projections, Eq. (8.69) implies that the components $\{\sigma[\kappa,\iota]\}$, $\{\sigma[\kappa,\bar{\iota}]\}$, $\{\sigma[\bar{\kappa},\iota]\}$, $\{\sigma[\bar{\kappa},\bar{\iota}]\}$ of the SHA spectrum are restored mainly from the coefficients arrays $\{\sigma_{1,0,0}[\kappa,\iota]\}$, $\{\sigma_{1,1,0}[\kappa,\iota]\}$, $\{\sigma_{1,0,1}[\kappa,\iota]\}$ and $\{\sigma_{1,1,1}[\kappa,\iota]\}$, respectively.

$$\begin{array}{l}\sigma_{1,0,0}[\kappa,\iota] \longleftrightarrow \sigma[\kappa,\iota],\\ \sigma_{1,1,0}[\kappa,\iota] \longleftrightarrow \sigma[\kappa,\bar{\iota}],\\ \sigma_{1,0,1}[\kappa,\iota] \longleftrightarrow \sigma[\bar{\kappa},\iota],\\ \sigma_{1,1,1}[\kappa,\iota] \longleftrightarrow \sigma[\bar{\kappa},\bar{\iota}],\end{array} \quad \kappa,\iota = -N/4,\ldots,N/4-1.$$

8.5.2.2 2D Transforms to Coarser Levels

Further transforms can be implemented either in a wavelet mode or in a wavelet packet mode.

2D wavelet transform

The M-level wavelet transform of a spline $S(x,y) \in {}^{2r,2r}\mathcal{S}$ presented in Eq. (8.58) results in decomposition of the spline into the orthogonal sum

$$\begin{aligned}S(x,y) = {} & S_{M,0,0}(x,y) + S_{M,1,0}(x,y) + S_{M,0,1}(x,y) + S_{M,1,1}(x,y)\\ & + S_{M-1,1,0}(x,y) + S_{M-1,0,1}(x,y) + S_{M-1,1,1}(x,y) + \ldots\\ & + S_{1,1,0}(x,y) + S_{1,0,1}(x,y) + S_{1,1,1}(x,y).\end{aligned}$$

The splines $S_{m,l,\tilde{l}}(x,y)$ are represented either via the orthonormal bases of exponential splines as in Eq. (8.59) or via the orthonormal basis of shifts of the wavelet packets $\psi^{2r}_{m,l,\tilde{l}}(x,y)$, $l,\tilde{l}=0,1$, as in Eq. (8.61). The coordinates in two representations are linked by Eq. (8.62).

The coordinates $\sigma_{m,0,0}[\kappa,\iota]$, $\sigma_{m,1,0}[\kappa,\iota]$, $\sigma_{m,0,1}[\kappa,\iota]$, $\sigma_{m,1,1}[\kappa,\iota]$ of the projections of the spline $S(x,y) \in {}^{2r,2r}\mathcal{S}$ onto the subspaces ${}^{2r,2r}\mathcal{W}_{m,0,0}$, ${}^{2r}\mathcal{W}_{m,1,0}$, ${}^{2r}\mathcal{W}_{m,1,0}$ and ${}^{2r}\mathcal{W}_{m,1,1}$, respectively, are derived from the coordinates $\sigma_{m-1,0,0}[\kappa,\iota]$ of the spline $S_{m-1,0,0}(x,y) \in {}^{2r,2r}\mathcal{W}_{m-1,0,0}$ in a way similar to Eq. (8.68):

$$\begin{pmatrix}\sigma_{m,0,0}[\kappa,\iota]\\ \sigma_{m,1,0}[\kappa,\iota]\\ \sigma_{m,0,1}[\kappa,\iota]\\ \sigma_{m,1,1}[\kappa,\iota]\end{pmatrix} = \frac{1}{2}\mathbf{A}_{m-1}[\kappa,\iota]\cdot\begin{pmatrix}\sigma_{m-1,0,0}[\kappa,\iota]\\ \sigma_{m-1,0,0}[\kappa,\bar{\iota}]\\ \sigma_{m-1,0,0}[\bar{\kappa},\iota]\\ \sigma_{m-1,0,0}[\bar{\kappa},\bar{\iota}]\end{pmatrix}, \tag{8.70}$$

$\kappa,\iota = -N_m/2,\ldots,N_m/2-1$, and $\mathbf{A}_{m-1}[\kappa,\iota] \stackrel{\text{def}}{=} \mathbf{A}_{m-1}[\kappa]\bigotimes\mathbf{A}_{m-1}[\iota]$. Here, $\bar{\kappa} \stackrel{\text{def}}{=} \kappa + N_m/2$, $\bar{\iota} \stackrel{\text{def}}{=} \iota + N_m/2$. The matrices $\mathbf{A}_{m-1}[n]$ are defined in Eq. (8.27).

The reconstruction of $S(x,y)$ is implemented iteratively, starting from the bottom level M similarly to Eq. (8.69):

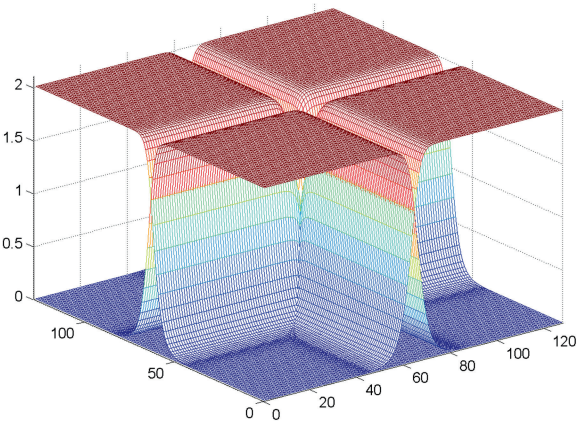


Fig. 8.19 Magnitudes of the SHA spectra of the wavelets of the first decomposition level: $\psi^{10}_{1,0,0}(x, y)$, $\psi^{10}_{1,1,0}(x, y)$, $\psi^{10}_{1,0,1}(x, y)$, $\psi^{10}_{1,1,1}(x, y)$

$$\begin{pmatrix} \sigma_{m-1,0,0}[\kappa, \iota] \\ \sigma_{m-1,0,0}[\kappa, \bar{\iota}] \\ \sigma_{m-1,0,0}[\bar{\kappa}, \iota] \\ \sigma_{m-1,0,0}[\bar{\kappa}, \bar{\iota}] \end{pmatrix} = 2\mathbf{A}^{\dagger}_{m-1}[\kappa, \iota] \cdot \begin{pmatrix} \sigma_{m,0,0}[\kappa, \iota] \\ \sigma_{m,1,0}[\kappa, \iota] \\ \sigma_{m,0,1}[\kappa, \iota] \\ \sigma_{m,1,1}[\kappa, \iota] \end{pmatrix}, \tag{8.71}$$

$\kappa, \iota = -N_m/2, \ldots, N_m/2 - 1$. Application of the transforms (8.71) can produce the SHA spectra of the projections splines and, in particular, the SHA spectra of the wavelets $\psi^{2r}_{m,0,0}(x, y)$ and $\psi^{2r}_{m,l,\tilde{l}}(x, y)$, $(l, \tilde{l}) = (1, 0), (0, 1)(1, 1)$. These SHA spectra provide a logarithmic-wise partition of the SHA spectral square $\kappa, \iota = -N/2, \ldots, N/2 - 1$.

Figure 8.19, which was prepared by the MATLAB code `mesh_wp_listP.m`, displays the magnitudes of the SHA spectra of the spline wavelets of order 10 from the first decomposition level.

Figure 8.17 displays the magnitudes of the SHA spectra of the wavelets for the 3 decomposition levels: $\psi^{10}_{3,0,0}$, $\left\{\psi^{10}_{3,l,\tilde{l}}\right\}$, $\left\{\psi^{10}_{2,l,\tilde{l}}\right\}$, $\left\{\psi^{10}_{1,l,\tilde{l}}\right\}$, where $(l, \tilde{l}) = (1, 0), (0, 1)(1, 1)$.

2D wavelet packet transform

The M-level wavelet packet transform of a spline $S(x, y) \in {}^{2r,2r}\mathscr{S}$ presented in Eq. (8.58) results in the decomposition of the spline into M orthogonal sums

$$S(x, y) = \bigoplus_{l,\tilde{l}=0}^{1} S_{1,l,\tilde{l}}(x, y) = \bigoplus_{l,\tilde{l}=0}^{3} S_{2,l,\tilde{l}}(x, y) = \ldots = \bigoplus_{l,\tilde{l}=0}^{2^M-1} S_{M,l,\tilde{l}}(x, y).$$

The splines $S_{m,l,\tilde{l}}(x, y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ are represented either via the orthonormal bases of exponential splines as in Eqs. (8.59) or via the orthonormal basis of shifts of

the wavelets $\varphi_m^{2r}(x, y)$ and $\psi^{2r}_{m,l,\tilde{l}}(x, y)$ as in Eq. (8.61). The coordinates in two representations are linked by Eq. (8.62).

The coordinates $\sigma_{m,0,0}[\kappa, \iota], \sigma_{m,1,0}[\kappa, \iota], \sigma_{m,0,1}[\kappa, \iota]$ and $\sigma_{m,1,1}[\kappa, \iota]$ of the splines $S(x, y)_{m,0,0}$, $S_{m,1,0}(x, y)$, $S_{m,0,1}(x, y)$ and $S_{m,1,1}(x, y)$, respectively, are derived from the coordinates $\sigma_{m-1,0,0}[\kappa, \iota]$ of the spline $S_{m-1,0,0}(x, y) \in {}^{2r,2r}\mathscr{W}_{m-1,0,0}$, using the matrices $\mathbf{A}_{m-1}[\kappa, \iota]$ as in Eq. (8.70). The same matrices are utilized to derive the coordinates $\sigma_{m,2l,2\tilde{l}}[\kappa, \iota], \sigma_{m,2l+1,2\tilde{l}}[\kappa, \iota], \sigma_{m,2l,2\tilde{l}+1}[\kappa, \iota]$ and $\sigma_{m,2l+1,2\tilde{l}+1}[\kappa, \iota]$ of the splines $S(x, y)_{m,2l,2\tilde{l}}$, $S_{m,2l+1,2\tilde{l}}(x, y)$, $S_{m,2l,2\tilde{l}+1}(x, y)$ and $S_{m,2l+1,2\tilde{l}+1}(x, y)$, respectively, from the coordinates $\sigma_{m-1,l,\tilde{l}}[\kappa, \iota]$ of the spline $S_{m-1,l,\tilde{l}}(x, y) \in {}^{2r}\mathscr{W}_{m-1,l,\tilde{l}}$. Taking into account the interchanging scheme described in Eq. (8.44), we get

$$\begin{pmatrix} \sigma_{m,\mathbf{l}_1}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_2}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_3}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_4}[\kappa, \iota] \end{pmatrix} = \frac{1}{2}\mathbf{A}_{m-1}[\kappa, \iota] \cdot \begin{pmatrix} \sigma_{m-1,l,\tilde{l}}[\kappa, \iota] \\ \sigma_{m-1,l,\tilde{l}}[\kappa, \bar{\iota}] \\ \sigma_{m-1,l,\tilde{l}}[\bar{\kappa}, \iota] \\ \sigma_{m-1,l,\tilde{l}}[\bar{\kappa}, \bar{\iota}] \end{pmatrix}, \tag{8.72}$$

where $\kappa, \iota = -N_m/2, \ldots, N_m/2 - 1$, and $\mathbf{A}_{m-1}[\kappa, \iota] \stackrel{\text{def}}{=} \mathbf{A}_{m-1}[\kappa] \bigotimes \mathbf{A}_{m-1}[\iota]$, the matrices $\mathbf{A}_{m-1}[n]$ are defined in Eq. (8.27) and

$$(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4) \tag{8.73}$$
$$= \begin{cases} ((2l, 2\tilde{l}), (2l+1, 2\tilde{l}), (2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}+1)), & \text{if } l \text{ even}, \tilde{l} \text{ even}; \\ ((2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}), (2l+1, 2\tilde{l})), & \text{if } l \text{ even}, \tilde{l} \text{ odd}; \\ ((2l, 2\tilde{l}+1), (2l, 2\tilde{l}), (2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}+1)), & \text{if } l \text{ odd}, \tilde{l} \text{ even}; \\ ((2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}), (2l, 2\tilde{l})), & \text{if } l \text{ odd}, \tilde{l} \text{ odd}. \end{cases}$$

The transforms of a 2D array are implemented by the MATLAB function `spl_WP_analP2D.m`. If the array consists of grid samples of a spline then the transform of the spline's coordinates is implemented by the MATLAB function `samp_spl_WP_analP2D.m`. The reconstruction of $S(x, y)$ is implemented iteratively, starting from any level m, similarly to Eq. (8.69):

$$\begin{pmatrix} \sigma_{m-1,l,\tilde{l}}[\kappa, \iota] \\ \sigma_{m-1,l,\tilde{l}}[\kappa, \bar{\iota}] \\ \sigma_{m-1,l,\tilde{l}}[\bar{\kappa}, \iota] \\ \sigma_{m-1,l,\tilde{l}}[\bar{\kappa}, \bar{\iota}] \end{pmatrix} = 2\mathbf{A}^{\dagger}_{m-1}[\kappa, \iota] \cdot \begin{pmatrix} \sigma_{m,\mathbf{l}_1}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_2}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_3}[\kappa, \iota] \\ \sigma_{m,\mathbf{l}_4}[\kappa, \iota] \end{pmatrix}, \tag{8.74}$$

where $\kappa, \iota = -N_m/2, \ldots, N_m/2 - 1$, and the multi-indices $(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4)$ are defined by Eq. (8.73). Application of the transforms (8.74) can produce the SHA spectra of the projections splines $S_{m,l,\tilde{l}}(x, y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ and, in particular, the SHA spectra of the wavelet packets $\psi^{2r}_{m,l,\tilde{l}}(x, y)$. These SHA spectra provide a variety of partitions of the SHA spectral square $\kappa, \iota = -N/2, \ldots, N/2 - 1$. Consequently, the shifts of

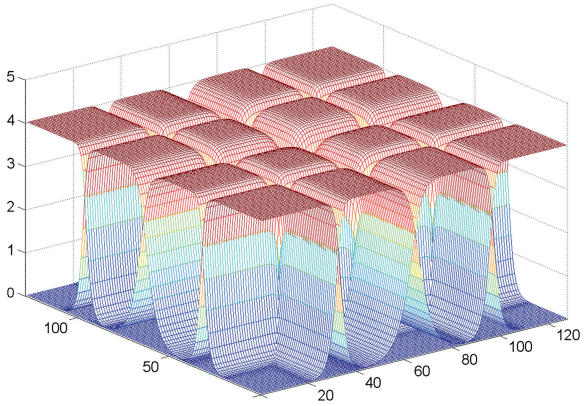


Fig. 8.20 Magnitudes of the SHA spectra of the spline wavelet packets $\psi^8_{2,l,\tilde{l}}(x, y), l, \tilde{l} = 0, 1, 2, 3$, of eight order from the second decomposition level

the wavelet packets provide a variety of the orthonormal bases for the spline space ${}^{2r,2r}\mathscr{S}$. As in the 1D case, the special cases are

- "Horizontal" bases that are bases consisting of shifts of all the wavelet packets from the same level m, where m can be one of the integers $1, \ldots, M$,

$$\mathbf{B}_m^{2h} \stackrel{\text{def}}{=} \bigcup_{l\tilde{l}=0}^{2^m-1} \left\{\psi^{2r}_{m,1,\tilde{l}}\left(x - 2^m k, y - 2^m n\right)\right\}, \tag{8.75}$$

  where $k, n = 0, \ldots, N_m - 1$, $N_m = N/2^m$. In this case, the SHA spectrum of a spline $S(x, y)$ is split into 2^{2m} blocks of the same size as, for example, in Fig. 8.19, where $m = 1$. Figure 8.20 (MATLAB code `mesh_wp_listP.m`) displays the magnitudes of the SHA spectra of the spline wavelet packets $\psi^8_{2,l,\tilde{l}}(x, y)$, $l, \tilde{l} = 0, 1, 2, 3$, of eight order from the second decomposition level. Their 4-sample shifts constitute an orthonormal basis for the spline space ${}^{8,8}\mathscr{S}$. For clarity, only the first quadrant $\kappa, \iota = 0, \ldots, N/2 - 1$, of the spectral square is displayed. The magnitudes of SHA spectra in the other quadrants mirror those in the first quadrant.
- Wavelet bases

$$\mathbf{B}_m^{2w} \stackrel{\text{def}}{=} \left\{\psi^{2r}_{m,0,0}\left(x - 2^m k, y - 2^m n\right)\right\}_{k,n=0}^{N_m-1} \bigcup_{\mu=m}^{1} [\left\{\psi^{2r}_{\mu,1,0}\left(x - 2^\mu k, y - 2^\mu n\right)\right\}$$

$$\bigcup \left\{\psi^{2r}_{\mu,0,1}\left(x - 2^\mu k, y - 2^\mu n\right)\right\} \bigcup \left\{\psi^{2r}_{\mu,1,1}\left(x - 2^\mu k, y - 2^\mu n\right)\right\}_{k,n=0}^{N_\mu-1}].$$

The SHA spectra of the wavelets from the first and third decomposition levels are displayed in Figs. 8.19 and 8.17, respectively.

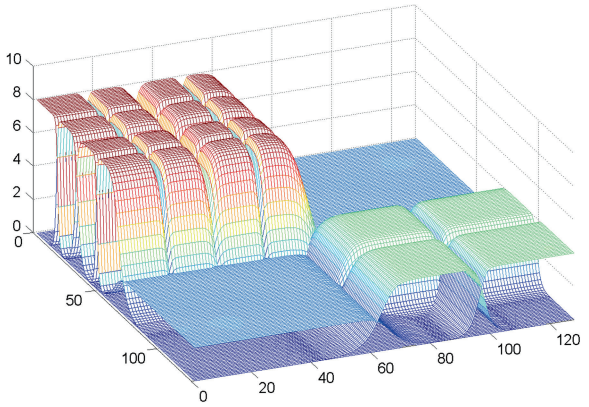


Fig. 8.21 Magnitudes of the SHA spectra of the eight-order wavelet packets for the 3 levels decomposition: $\psi^8_{3,0,0}(x,y)$, $\psi^8_{3,1,0}(x,y)$, $\psi^8_{3,0,1}(x,y)$, $\left\{\psi^8_{3,l,\tilde{l}}(x,y)\right\}, l,\tilde{l}=1,2,3$, $\left\{\psi^8_{2,l,\tilde{l}}(x,y)\right\}$, $l,\tilde{l}=2,3$, $\psi^8_{1,1,0}(x,y)$ and $\psi^8_{1,0,1}(x,y)$

Besides these two options, any combination of wavelet packets, such that their SHA spectra form a one-fold cover of the whole spectral domain $\kappa, \iota = -N/2, \ldots N/2-1$, generate an orthonormal basis for the spline space ${}^{2r,2r}\mathscr{S}$.

Figure 8.21 (MATLAB code `mesh_wp_listP.m`) displays the magnitudes of the SHA spectra of a combination of eight order wavelet packets, whose shifts constitute an orthonormal basis for ${}^{8,8}\mathscr{S}$. These are the wavelet packets $\psi^8_{3,0,0}(x,y)$, $\psi^8_{3,1,0}(x,y)$, $\psi^8_{3,0,1}(x,y)$, $\left\{\psi^8_{3,l,\tilde{l}}(x,y)\right\}$, $l,\tilde{l}=1,2,3$, from the third decomposition level, $\left\{\psi^8_{2,l,\tilde{l}}(x,y)\right\}$, $l,\tilde{l}=2,3$ from the second decomposition level and $\psi^8_{1,1,0}(x,y)$, $\psi^8_{1,0,1}(x,y)$ from the first decomposition level. We observe that their SHA spectra cover the whole spectral domain.

The above figures were prepared by the MATLAB code `mesh_wp_listP.m`. The direct and inverse wavelet transforms of the 2D spline coordinates are implemented by the MATLAB functions `spl_Wv_analP_2D` and `spl_Wv_synthP_2D`, respectively.

8.5.3 Best Basis

Once the wavelet packet transform of a spline $S(x,y) \in {}^{2r,2r}\mathscr{S}$ is implemented, a variety of orthonormal wavelet packet bases becomes available. One can select from this variety the best, with respect to a certain additive cost function $C(\mathbf{B})$, basis for representation of the spline $S(x,y)$. For this, we utilize a scheme, which is similar to the 1D scheme described in Sect. 8.4.4. As in the 1D case, the 2D wavelet packet transform has a tree structure. The difference is that a space ${}^{2r}\mathscr{W}_{m-1,l,\tilde{l}}$ is split into

four orthogonal subspaces as shown in Eq. (8.60):

$$ {}^{2r}\mathscr{W}_{m-1,l,\tilde{l}} = {}^{2r}\mathscr{W}_{m,2l,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{m,2l,2\tilde{l}+1} \bigoplus {}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}+1}. $$

Thus, the 2D wavelet packet transforms form quad-trees. The following statement is a counterpart of Proposition 8.12.

Proposition 8.13 [4] *Let $\mathbf{B}_{m-1,l,\tilde{l}}$ be a 2D wavelet packet orthonormal basis for the projection of a spline $S(x, y) \in {}^{2r,2r}\mathscr{S}$ onto the space ${}^{2r}\mathscr{W}_{m-1,l,\tilde{l}}$ and $\bar{\mathbf{B}}_{m,2l,2\tilde{l}}$, $\bar{\mathbf{B}}_{m,2l+1,2\tilde{l}}$, $\bar{\mathbf{B}}_{m,2l,2\tilde{l}+1}$ and $\bar{\mathbf{B}}_{m,2l+1,2\tilde{l}+1}$ be the best bases for the subspaces ${}^{2r}\mathscr{W}_{m,2l,2\tilde{l}}$, ${}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}}$, ${}^{2r}\mathscr{W}_{m,2l,2\tilde{l}+1}$ and ${}^{2r}\mathscr{W}_{m,2l+1,2\tilde{l}+1}$, respectively. If C is an additive cost function then the best basis for the space ${}^{2r}\mathscr{W}_{m-1,l,\tilde{l}}$*

$$ \begin{aligned} &\bar{\mathbf{B}}_{m-1,l} = \mathbf{B}_{m-1,l,\tilde{l}} \quad \text{if} C(\mathbf{B}_{m-1,l,\tilde{l}}) < \textstyle\sum_{\lambda.\mu=0,1} C(\mathbf{B}_{m,2l+\lambda,2\tilde{l}+\mu}); \\ &\bar{\mathbf{B}}_{m-1,l} = \bar{\mathbf{B}}_{m,2l,2\tilde{l}} \bigcup \bar{\mathbf{B}}_{m,2l+1,2\tilde{l}} \bigcup \bar{\mathbf{B}}_{m,2l,2\tilde{l}+1} \bigcup \bar{\mathbf{B}}_{m,2l+1,2\tilde{l}+1} \quad \text{otherwise.} \end{aligned} \tag{8.76} $$

Design of the best wavelet packet basis for a spline $S(x, y) \in {}^{2r,2r}\mathscr{S}$ consists of the following steps:

1. Implement wavelet packet transform of the spline $S((x, y)$ down to a level M.
2. Calculate the cost function

$$ C_{m,l,\tilde{l}} \stackrel{\text{def}}{=} C\left(\left\{\psi^{2r}_{m,l,\tilde{l}}\left(x - 2^m k, y - 2^m n\right)\right\}\right) $$

   for all the bases of the subspaces ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$, $m = 1, \ldots M$, of the space ${}^{2r,2r}\mathscr{S}$.
3. The best (and the single available) basis $\bar{\mathbf{B}}^M$ for the bottom level M is the union of the wavelet packet bases of the subspaces ${}^{2r}\mathscr{W}_{M,l,\tilde{l}}$, $l, \tilde{l} = 0, \ldots, 2^M - 1$.
4. The best basis $\bar{\mathbf{B}}^{M-1}$ for the upper level $M - 1$ is designed in line with the rule Eq. (8.76) by comparing the cost functions of the "parent" subspaces with ${}^{2r}\mathscr{W}_{M-1,l,\tilde{l}}$ with the cost functions of their "offsprings" ${}^{2r}\mathscr{W}_{M,2l,2\tilde{l}}$, ${}^{2r}\mathscr{W}_{M,2l+1,2\tilde{l}}$, ${}^{2r}\mathscr{W}_{M,2l+1,2\tilde{l}}$ and ${}^{2r}\mathscr{W}_{M,2l+1,2\tilde{l}+1}$.
5. The procedure is iterated up to the top space ${}^{2r,2r}\mathscr{S}$.

Once the wavelet packet transform of a signal is implemented and the entropies in each block of the transform coefficients are calculated by the MATLAB function `spl_WP_analP2D.m` or `samp_spl_WP_analP2D.m`, the list of the wavelet packets, whose shifts constitute the best basis tree is derived by the function `WP_BB_Listree2_D_P.m`.

The MATLAB function `List_synthP_2D_samp.m` restores the spline $S(x, y)$ from the basis defined by a list Λ. It produces the coefficients of the spline $\{q[k, n]\}$ in the orthonormal basis $\left\{\varphi^{2r}(x - k, y - n)\right\}$, the SHA spectrum of the spline and its grid samples.

References

1. A. Averbuch, E. Hulata, V. Zheludev, I. Kozlov, A wavelet packet algorithm for classification and detection of moving vehicles. Multidimension. Syst. Signal Process. **12**(1), 9–31 (2001)
2. G. Battle, A block spin construction of ondelettes. I. lemarié functions. Comm. Math. Phys. **110**(4), 601–615 (1987)
3. C.K. Chui, J.-Z. Wang, On compactly supported spline wavelets and a duality principle. Trans. Amer. Math. Soc. **330**(2), 903–915 (1992)
4. R.R. Coifman, V.M. Wickerhauser, Entropy-based algorithms for best basis selection. IEEE Trans. Inform. Theory **38**(2), 713–718 (1992)
5. R.A. Horn, C.R. Johnson, *Topics in Matrix Analysis* (Cambridge University Press, Cambridge, 1994)
6. P.G. Lemarié. Ondelettes à localisation exponentielle. J. Math. Pures Appl. (9), **67**(3):227–236 (1988)
7. P. Neittaanmäki, V. Rivkind, V. Zheludev, in *Periodic Spline Wavelets and Representation of Integral Operators*, Preprint 177, University of Jyväskylä, Department of Mathematics, 1995
8. G. Plonka, M. Tasche, On the computation of periodic spline wavelets. Appl. Comput. Harmon. Anal. **2**(1), 1–14 (1995)
9. N. Saito, R.R. Coifman, in *Improved Discriminant Bases Using Empirical Probability Density Estimation*. Proceedings of the Statistical Computing Section of Amer. Statist. Assoc., (Washington, DC, 1997), pp. 312–321
10. N. Saito, R.R. Coifman, Local discriminant bases and their applications. J. Math. Imaging Vision **5**(4), 337–358 (1995)
11. C.E. Shannon, W. Weaver, *The Mathematical Theory of Communication* (The University of Illinois Press, Urbana, IL, 1949)
12. J.-O. Strömberg, in *A Modified Franklin System and Higher-Order Spline Systems of R^n as Unconditional Bases for Hardy Spaces*. Conference on Harmonic Analysis in Honor of Antoni Zygmund, Vol. I, II (Chicago, Ill., 1981), Belmont, 1983, pp. 475–494
13. M. Unser, A. Aldroubi, M. Eden, A family of polynomial spline wavelet transforms. Signal Process. **30**(2), 141–162 (1993)
14. M.V. Wickerhauser, *Adapted Wavelet Analysis: From Theory to Software* (AK Peters, Wellesley, MA, 1994)
15. V. Zheludev. *Periodic Splines, Harmonic Analysis, and Wavelets*, ed. by Y.Y. Zeevi, R. Coifman. Signal and Image Representation in Combined Spaces, volume 7 of Wavelet Anal. Appl. (Academic Press, San Diego, CA, 1998) pp. 477–509
16. V. Zheludev, Wavelets based on periodic splines. Russian Acad. Sci. Dokl. Math. **49**(2), 216–222 (1994)

Chapter 9
Discrete-Time Periodic Wavelet Packets

Abstract Direct and inverse wavelet and wavelet packet transforms of a spline are implemented by filtering the spline's coordinates by two-channel critically sampled p-filter banks. In this chapter, those p-filter banks are utilized for processing discrete-time signals. The p-filter banks generate discrete-time wavelets and wavelet packets in the spaces of 1D and 2D periodic signals.

9.1 One-Dimensional Periodic Discrete-Time Wavelet Packets

Actually, wavelet packet transform of a periodic spline reduces to application of a two-channel p-filter bank with downsampling factor of 2 to the coordinates of the spline in the orthonormal basis. We outline the spline transform scheme from that point of view.

9.1.1 Summary of One-Level Wavelet Transform of a Spline

Assume a spline $S(t) \in {}^{2r}\tilde{\mathscr{S}}$ is represented by the two orthonormal bases

$$S(t) = \sum_{k=0}^{N-1} q[k]\,\varphi^{2r}(t-k), \quad = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\gamma^{2r}[n](t), \tag{9.1}$$

where $\varphi^{2r}(t)$ is the Battle–Lemarié spline and $\gamma^{2r}[n](t)$ are the orthonormal exponential splines. The coordinates are linked via the DFT:

$$\sigma[n] = \hat{q}[n] = \sum_{k=0}^{N-1} \omega^{-nk}\, q[k], \quad q[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \sigma[n].$$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_9,

It was described in Chap. 8 that the one-level wavelet transform of the spline $S(t)$ results in splitting the spline into the orthogonal sum

$$S(t) = S_{1,0}(t) \bigoplus S_{1,1}(t), \quad S_{1,p}(t) = \sum_{k=0}^{N/2-1} q_{1,p}[k]\, \psi_{1,p}^{2r}(t-2k)$$
$$= \sqrt{\frac{1}{2N}} \sum_{n=-N/4}^{N/4-1} \sigma_{1,p}[n]\, \gamma_{1,p}^{2r}[n](t), \quad q_{1,p}[k] = \frac{2}{N} \sum_{n=-N/4}^{N/4-1} \omega^{2nk}\, \sigma_{1,p}[n], \ p = 0, 1.$$

The coordinates are linked as follows

$$\sigma_{1,0}[n] = \sqrt{\frac{1}{2}}\ (s_0[n]\, \sigma[n] + s_0[n+N/2]\, \sigma[n+N/2]),$$
$$\sigma_{1,1}[n] = \sqrt{\frac{1}{2}}\ (d_0[n]\, \sigma[n](t) + d_0[n+N/2]\, \sigma[n+N/2]),$$
$$s_0[n] = \sqrt{\frac{u^{4r}[n]}{2\, u_1^{4r}[n]}}\ \cos^{2r}\frac{\pi n}{N}, \quad d_0[n] = \omega^n \sqrt{\frac{u^{4r}[n+N/2]}{2\, u_1^{4r}[n]}}\ \sin^{2r}\frac{\pi n}{N}.$$

Recall that $\sqrt{2}\, s_0[n] = \nu_{1,0}[n]$ and $\sqrt{2}\, d_0^*[n] = \nu_{1,1}[n]$, where $\nu_{1,p}[n]$ are the SHA spectra of the Battle–Lemarié wavelets $\psi_{1,p}(t)$, $p = 0, 1$ of the first level. Then, we have

$$\begin{aligned} q_{1,0}[k] &= \frac{2}{N} \sum_{n=0}^{N/2-1} \omega^{2nk}\, \sigma_{1,0}[n] \\ &= \frac{\sqrt{2}}{N} \sum_{n=0}^{N/2-1} \omega^{2nk}\ (s_0[n]\, \sigma[n] + s_0[n+N/2]\, \sigma[n+N/2]) \qquad (9.2) \\ &= \frac{\sqrt{2}}{N} \sum_{n=0}^{N-1} \omega^{2nk}\, s_0[n]\, \sigma[n]. \end{aligned}$$

Denote $\mathbf{h}_0^0 = \left\{h_0^0[k]\right\}$ and $\mathbf{h}_0^1 = \left\{h_0^1[k]\right\}$ the p-filters in the space $\Pi[N]$, whose frequency responses are

$$\hat{h}_0^0[n] = \sqrt{2}\, s_0[n] = \nu_{1,0}[n], \quad \hat{h}_0^1[n] = \sqrt{2}\, d_0^*[n] = \nu_{1,1}[n] \qquad (9.3)$$
$$\Longrightarrow h_0^0[k] = \frac{\sqrt{2}}{N} \sum_{n=0}^{N-1} \omega^{nk}\, s_0[n], \quad h_0^1[k] = \frac{\sqrt{2}}{N} \sum_{n=0}^{N-1} \omega^{nk}\, d_0^*[n].$$

Then, Eq. (9.2) implies that

$$q_{1,0}[k] = \sum_{l=0}^{N-1} h_0^0[2k-l]\, q[l] = \sum_{l=0}^{N-1} h_0^0[l-2k]\, q[l]. \tag{9.4}$$

Similarly, we get

$$q_{1,1}[k] = \frac{\sqrt{2}}{N} \sum_{n=0}^{N/2-1} \omega^{2nk}\, d_0[n]\, \sigma[n] = \frac{1}{N} \sum_{n=0}^{N/2-1} \omega^{2nk}\, (\hat{h}_0^1[n])^*\, \sigma[n]$$

$$\Longrightarrow q_{1,1}[k] = \sum_{l=0}^{N-1} h_0^1[l-2k]\, q[l]. \tag{9.5}$$

Equations (9.4) and (9.5) mean that the transform coefficients $\{q_{1,0}[k]\}$ and $\{q_{1,1}[k]\}$, $k = 0, ..., N/2-1$, are the result of application of the two-channel analysis p-filter bank $\mathbf{H} = \mathbf{h}_0^0 \bigcup \mathbf{h}_0^1$ with downsampling factor of 2, to the vector $\{q[k]\}$, $k = 0, ..., N-1$, of the spline coordinates.

The inverse transform of coordinates is $\sigma[n] = \sigma^0[n] + \sigma^1[n]$, where

$$\sigma^0[n] = \sqrt{2} s_0[n]\, \sigma_{1,0}[n] = \hat{h}_0^0[n]\, \sigma_{1,0}[n], \quad \sigma^1[n] = \sqrt{2} d_0^*[n]\, \sigma_{1,1}[n] = \hat{h}_0^1[n]\, \sigma_{1,1}[n].$$

Then, for $p = 0, 1$, the coefficients $q^p[k]$ are

$$q^p[k] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{nk}\, \hat{h}_0^p[n]\, \sigma_{1,p}[n] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{nk} \sum_{m=0}^{N-1} \omega^{-nm}\, h_0^p[m] \sum_{l=0}^{N/2-1} \omega^{-2nl}\, q_{1,p}[l]$$

$$= \sum_{l=0}^{N/2-1}, \sum_{s=0}^{N-1} h_0^p[s]\, q_{1,p}[l]\, \frac{1}{N} \sum_{n=0}^{N-1} \omega^{n(k-s-2l)}) = \sum_{l=0}^{N/2-1} h_0^p[k-2l]\, q_{1,p}[l].$$

Hence, in order to recover the vector $\mathbf{q}$ of the spline coordinates, the vectors $\mathbf{q}_{1,0}$ and $\mathbf{q}_{1,1}$, which are upsampled by factor of 2, are filtered by the p-filters $\mathbf{h}_0^0$ and $\mathbf{h}_0^1$, respectively:

$$q[k] = q^0[k] + q^1[k] = \sum_{l=0}^{N/2-1} h_0^0[k-2l]\, q_{1,0}[l] + h_0^1[k-2l]\, q_{1,1}[l]. \tag{9.6}$$

The diagram in Fig. 9.1 illustrates the direct and inverse transforms. The notation $\overleftarrow{h}_0^p$, $p = 0, 1$, means application of the p-filters in a reverse order.

The magnitude responses (MR) $\hat{h}_0^0[n] = \sqrt{2} s_0[n]$ and $|\hat{h}_0^1[n]| = \sqrt{2} s_0[n + N/2]$ of p-filters for splines of different orders are displayed in Fig. 8.2. We observe that the MR mirror each other, their passbands and stopbands are flat and their shapes tend to rectangular as the spline order increases. It will be shown in the next section that impulse responses are symmetric and well localized in time domain. It is only natural to utilize these p-filters for signal processing purposes.

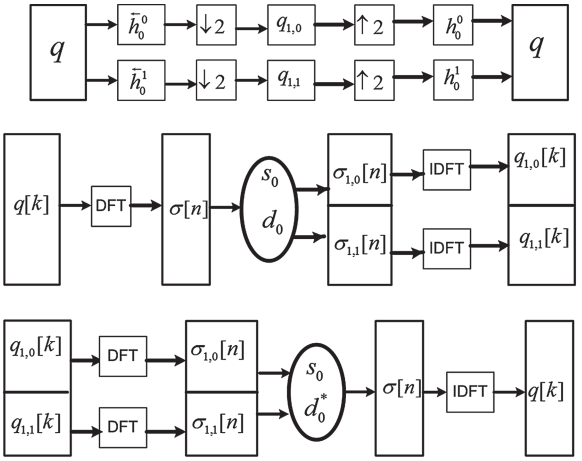


Fig. 9.1 *Top* Block scheme of the direct and inverse transforms of the coordinates $\mathbf{q} = \{q[k]\}_{k=0}^{N-1}$ of a periodic spline in the orthonormal basis $\left\{\varphi^{2r}(t-k)\right\}_{k=0}^{N-1}$. *Center* Diagram of practical implementation of the direct transform. *Bottom* Diagram of implementation of the inverse transform

9.1.2 One-Level Wavelet Transform of a Signal

Assume, a discrete-time signal $\mathbf{x} \stackrel{\text{def}}{=} \{x[k]\}$ belongs to $\Pi[N]$. That means that it is periodic with the period N, $N = 2^j$. The inner product, the norm and the circular convolution in $\Pi[N]$ are defined in Sect. 1.2.1. The signal $\mathbf{x}$ is transformed according to the scheme in Fig. 9.1.

9.1.2.1 Direct and Inverse Transforms

Denote

$$\sigma[n] \stackrel{\text{def}}{=} \hat{x}[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, x[k] \Longleftrightarrow x[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \sigma[n].$$

Transform the array $\{\sigma[n]\}$

$$\sigma_{1,p}[n] = \frac{1}{2}\left(\hat{h}_0^p[n])^*\, \sigma[n] + (\hat{h}_0^p[n+N/2])^*\, \sigma[n+N/2]\right), \quad p = 0, 1,$$

where the frequency responses $\hat{h}_0^p[n]$ are defined in Eq. (9.3). Apply the IDFT

$$\begin{aligned}\xi_{1,p}[l] &= \frac{2}{N}\sum_{n=0}^{N/2-1}\omega^{2ln}\,\sigma_{1,p}[n]\\ &= \frac{1}{N}\sum_{n=-0}^{N/2-1}\omega^{2ln}\left(\hat{h}_0^p[n])^*\,\sigma[n]+(\hat{h}_0^p[n+N/2])^*\,\sigma[n+N/2]\right) \quad (9.7)\\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sigma[n]\,\omega^{2ln}\,(\hat{h}_0^p[n])^* = \sum_{m=0}^{N-1}h_0^p[m-2l]\,x[m],\ \ p=0,1.\end{aligned}$$

Once the coefficients $\left\{\xi_{1,p}[k]\right\}$, $p = 0, 1,\ \ l = 0, ..., N/2 - 1$, are known, the original signal can be restored as

$$x[k] = \sum_{l=0}^{N/2-1} h_0^0[k-2l]\,\xi_{1,0}[l] + h_0^1[k-2l]\,\xi_{1,1}[l], \quad (9.8)$$

which is derived similarly to Eq. (9.6).

9.1.2.2 Discrete-Time Periodic Wavelets of the First Level

Denote the impulse responses of the p-filters $\mathbf{h}_0^p$, $p = 0, 1$, as

$$\begin{aligned}\Psi_{1,0}^{2r}[l] &\stackrel{\text{def}}{=} h_0^0[l] = \frac{\sqrt{2}}{N}\sum_{n=-N/2}^{N/2-1}\omega^{ln}\,s_0[n] = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\omega^{ln}\,\nu_{1,0}[n], \quad (9.9)\\ \Psi_{1,1}^{2r}[l] &\stackrel{\text{def}}{=} h_0^1[l] = \frac{\sqrt{2}}{N}\sum_{n=-N/2}^{N/2-1}\omega^{ln}\,d_0^*[n] = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\omega^{ln}\,\nu_{1,1}[n],\end{aligned}$$

where $\left\{\nu_{1,p}[n]\right\}$ are the SHA spectra of the wavelets $\psi_{1,p}^{2r}(t)$, and call them the discrete-time wavelets of the first decomposition level. Then, Eq. (9.7) can be presented as the inner product of $\mathbf{x}$ with the signal $\Psi_{1,p}^{2r}[\cdot - 2k] \stackrel{\text{def}}{=} \left\{\Psi_{1,p}^{2r}[l-2k]\right\}$, $l = 0, ...N - 1$:

$$\xi_{1,p}[k] = \sum_{l=0}^{N-1}\Psi_{1,p}^{2r}[l-2k]\,x[l] = \left\langle \mathbf{x},\, \Psi_{1,p}^{2r}[\cdot-2k]\right\rangle,\ \ p=0,1.$$

Consequently, Eq. (9.8) implies that any signal $\mathbf{x} \in \Pi[N]$ can be expanded as follows:

$$x[k] = \sum_{l=0}^{N/2-1} \Psi_{1,0}^{2r}[k-2l]\,\xi_{1,0}[l] + \Psi_{1,1}^{2r}[k-2l]\,\xi_{1,1}[l]$$

$$= \sum_{l=0}^{N/2-1} \left\langle \mathbf{x}, \Psi_{1,0}^{2r}[\cdot - 2l] \right\rangle \Psi_{1,0}^{2r}[k-2l] + \left\langle \mathbf{x}, \Psi_{1,1}^{2r}[\cdot - 2l] \right\rangle \Psi_{1,1}^{2r}[k-2l] \Longleftrightarrow$$

$$\mathbf{x} = \sum_{l=0}^{N/2-1} \left\langle \mathbf{x}, \Psi_{1,0}^{2r}[\cdot - 2l \right\rangle \Psi_{1,0}^{2r}[\cdot - 2l] + \left\langle \mathbf{x}, \Psi_{1,1}^{2r}[\cdot - 2l] \right\rangle \Psi_{1,1}^{2r}[\cdot - 2l]. \quad (9.10)$$

This relation implies the following fact.

Proposition 9.1 *The set of signals*

$$\mathbf{B}^1 \stackrel{\text{def}}{=} \left\{ \Psi_{1,0}^{2r}[\cdot - 2l] \right\} \bigcup \left\{ \Psi_{1,1}^{2r}[\cdot - 2l] \right\}, \quad l = 0, ..., N/2, \qquad (9.11)$$

from the space $\Pi[N]$ forms an orthonormal basis of this space.

Proof Any signal $\mathbf{x}$ from the N-dimensional space $\Pi[N]$ can be expanded over the system $\mathbf{B}^1$, which comprises N elements. Therefore, the elements of $\mathbf{B}^1$ are linearly independent and form a basis of $\Pi[N]$. Take $\mathbf{x} = \Psi_{1,0}^{2r}$. Then, it follows from Eq. (9.10) that

$$\Psi_{1,0}^{2r}[k] = \left\langle \Psi_{1,0}^{2r}, \Psi_{1,0}^{2r} \right\rangle \Psi_{1,0}^{2r}[k] + \sum_{l=1}^{N/2-1} \left\langle \Psi_{1,0}^{2r}, \Psi_{1,0}^{2r}[\cdot - 2l] \right\rangle \Psi_{1,0}^{2r}[k-2l]$$
$$+ \sum_{l=0}^{N/2-1} \left\langle \Psi_{1,0}^{2r}, \Psi_{1,1}^{2r}[\cdot - 2l] \right\rangle \Psi_{1,1}^{2r}[k-2l]$$

Due to linear independence of elements of $\mathbf{B}^1$, the inner products are

$$\left\langle \Psi_{1,0}^{2r}, \Psi_{1,p}^{2r}[\cdot - 2l] \right\rangle = 0, \text{ for } l = 1, ..., N/2 - 1, \quad \left\langle \Psi_{1,0}^{2r}, \Psi_{1,1}^{2r} \right\rangle = 0$$

and $\left\langle \Psi_{1,0}^{2r}, \Psi_{1,0}^{2r} \right\rangle = \left\| \Psi_{1,0}^{2r}, \Psi_{1,1}^{2r} \right\|^2 = 1.$

The same is true for $\Psi_{1,1}^{2r}$. ■

Properties of the signals $\Psi_{1,p}^{2r}$, $p = 0, 1$, are very similar to properties of the respective orthonormal periodic Battle-Lemarié wavelets $\psi_{1,p}^{2r}(t)$ of the first level. They are mutually orthogonal, their Euclidean norms are equal to one and their 2 sample circular shifts form an orthonormal basis of the space $\Pi[N]$ of periodic discrete-time signals. In addition, the DFT of the signals $\Psi_{1,p}^{2r}$ coincide with the SHA spectra of the wavelets $\psi_{1,p}^{2r}(t)$. Therefore, it is natural to call the signals $\Psi_{1,p}^{2r}$, $p = 0, 1$, discrete-time periodic wavelets of the first level. Figure 9.2 displays

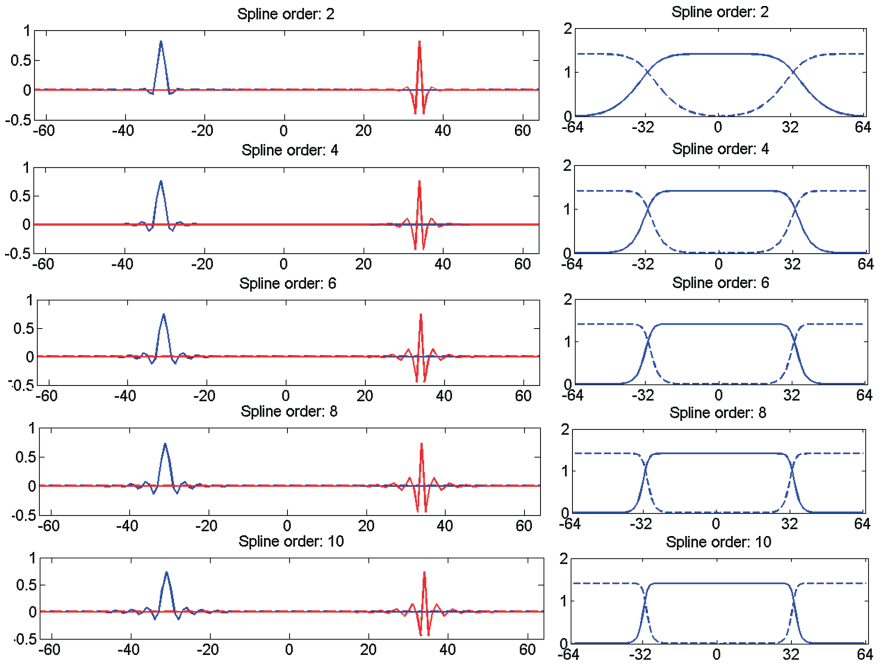


Fig. 9.2 *Left frames* discrete-time periodic wavelets $\Psi_{1,0}^{2r}$ (*left*) and $\Psi_{1,1}^{2r}$ (*right*), $r = 1, 2, 3, 4, 5$, $N = 64$. *Right frames* DFT of $\Psi_{1,0}^{2r}$ (*solid lines*) and of $\Psi_{1,1}^{2r}$ (*dashed lines*)

discrete-time periodic wavelets $\Psi_{1,p}^{2r}$, $p = 0, 1$, of even orders 2 to 10 and their DFT. Observe that the wavelets are symmetric and well localized in time domain. Their shape is, to some extent, similar to the shape of the respective wavelets $\psi_{1,p}^{2r}(t)$, which are depicted in Fig. 8.3. The figure was produced by the MATLAB code `Di_gen_wp_L1P.m`.

The discrete-time wavelets $\Psi_{1,0}^{2r}$ and $\Psi_{1,1}^{2r}$ generate a split of the signal space $\Pi[N]$ into two orthogonal subspaces $\Pi[N] = \Pi_{1,0}[N] \bigoplus \Pi_{1,1}[N]$, which consist of signals

$$\begin{aligned} \mathbf{x}_{1,0} &= \left\{ x_{1,0}[k] = \sum_{l=0}^{N/2-1} \xi_{1,0}[l]\, \Psi_{1,0}^{2r}[k-2l] \right\} \in \Pi_{1,0}[N], \\ \mathbf{x}_{1,1} &= \left\{ x_{1,1}[k] = \sum_{l=0}^{N/2-1} \xi_{1,1}[l]\, \Psi_{1,1}^{2r}[k-2l] \right\} \in \Pi_{1,1}[N], \end{aligned} \tag{9.12}$$

respectively. In order to orthogonally project a signal $\mathbf{x}$ onto the subspaces $\Pi_{1,0}[N]$ and $\Pi_{1,1}[N]$, the coefficients should be selected as

$$\begin{aligned} \xi_{1,0}[l] &= \left\langle \mathbf{x}, \Psi_{1,0}^{2r}[\cdot - 2l] \right\rangle = \sum_{k=0}^{N-1} x[k], \Psi_{1,0}^{2r}[k-2l], \\ \xi_{1,1}[l] &= \left\langle \mathbf{x}, \Psi_{1,1}^{2r}[\cdot - 2l] \right\rangle = \sum_{k=0}^{N-1} x[k], \Psi_{1,1}^{2r}[k-2l]. \end{aligned} \tag{9.13}$$

If the coefficients $\left\{\xi_{1,0}[l]\right\}$ and $\left\{\xi_{1,1}[l]\right\}$, $l = 0, ..., N/2-1$, are known, then, practically, the components $\mathbf{x}_{1,0}$ and $\mathbf{x}_{1,1}$ of the signal $\mathbf{x}$ can be computed via the DFT. Equations (9.9) and (9.12) imply that

$$\hat{x}_{1,p}[n] = \sum_{k=0}^{N-1} \omega^{-nk} \sum_{l=0}^{N/2-1} \xi_{1,0}[l]\, \Psi_{1,0}^{2r}[k-2l] = \sigma_{1,p}[n]\, \nu_{1,p}[n],$$

$$\sigma_{1,p}[n] = \sum_{l=0}^{N/2-1} \omega^{-2ln}\, \xi_{1,p}[l], \quad p = 0, 1.$$

Thus, for $k = 0, ...N-1$

$$x_{1,p}[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \sigma_{1,p}[n]\, \nu_{1,p}[n], \quad p = 0, 1.$$

Remark 9.1.1 The relations (9.13) mean that the coefficients $\xi_{1,p}[l]$, $p = 0, 1$, are the result of filtering the signal $\mathbf{x}$ by the p-filters, whose impulse responses are $\left\{\Psi_{1,p}^{2r}[-k]\right\}$, which is followed by downsampling by factor of 2. The relations (9.12) mean that the signals $\mathbf{x}_{1,p} = \left\{x_{1,p}[k]\right\}$, $p = 0, 1$, are the result of application of the p-filters, whose impulse responses are $\left\{\Psi_{1,p}^{2r}[k]\right\}$, to the sequences $\xi_{1,p}[l]$, which are upsampled by factor of 2, respectively.

The DFT of the signals $\mathbf{x}_{1,0} \in \Pi_{1,0}[N]$ and $\mathbf{x}_{1,1} \in \Pi_{1,1}[N]$ are localized within the bands, which are determined by the DFT of the discrete-time wavelets $\Psi_{1,0}^{2r}$ and $\Psi_{1,1}^{2r}$ (SHA spectra of the wavelets $\psi_{1,0}^{2r}(t)$ and $\psi_{1,1}^{2r}(t)$), respectively. These SHA spectra are effectively confined in the intervals $[-N/4, N/4]$ and $[-N/2, -N/4] \bigcup [N/4, N/2]$, respectively.

Due to orthonormality of shifts of the wavelets $\Psi_{1,0}^{2r}$ and $\Psi_{1,1}^{2r}$, the norms of signals are:

$$\left\|\mathbf{x}_{1,p}\right\|^2 = \sum_{l=0}^{N/2-1} \left|\xi_{1,p}[l]\right|^2, \; p = 0, 1, \quad \|\mathbf{x}\|^2 = \sum_{l=0}^{N/2-1} \left|\xi_{1,0}[l]\right|^2 + \left|\xi_{1,1}[l]\right|^2.$$

Loosely speaking, the sum $\sum_{l=0}^{N/2-1} \left|\xi_{1,0}[l]\right|^2$ characterizes the energy of the low-frequency component $\mathbf{x}_{1,0}$ of the signal $\mathbf{x}$, whose DFT is concentrated within the band $[-N/2, N/2]$. On the other hand, the sum $\sum_{l=0}^{N/2-1} \left|\xi_{1,1}[l]\right|^2$ characterizes the energy of the high-frequency component $\mathbf{x}_{1,1}$ of the signal $\mathbf{x}$, whose DFT is concentrated within the band $[-N/2, -N/4] \bigcup [N/4, N/2]$.

9.1.3 Multilevel Wavelet Packet Transforms of a Signal

In this section, the spline wavelet packet transforms defined in Sect. 8.3 are iteratively applied to signals from $\Pi[N]$. These operations produce a variety of splits of the signal space $\Pi[N]$ into sets of mutually orthogonal subspaces and so also a variety of discrete-time periodic signals, whose circular shifts form orthonormal bases of the subspaces.

Recall the notations

$$N_m = \frac{N}{2^m}, \quad s_{m-1}[n] = \sqrt{\frac{u_{m-1}^{4r}[n]}{2\,u_m^{4r}[n]}}\ \cos^{2r}\frac{\pi n}{N_{m-1}},$$

$$d_{m-1}[n] = \omega^{2^{m-1}n}\, s_{m-1}[n+N_m] = \omega^{2^{m-1}n}\sqrt{\frac{u_{m-1}^{4r}[n+N_m]}{2\,u_m^{4r}[n]}}\ \sin^{2r}\frac{\pi n}{N_{m-1}},$$

$$n = -N_m, ..., N_m - 1.$$

9.1.3.1 Direct Multilevel Transform

Assume $\mathbf{x} = \{x[k]\}$ is a signal from $\Pi[N]$. Apply the multilevel wavelet packet transforms defined by Eq. (8.44) to the sequence $\{\sigma[n] = \hat{x}[n]\}$ down to level M. As a result, a collection of sequences is derived:

$$\{\sigma_{1,0}[n]\} \bigcup \{\sigma_{1,1}[n]\},\ n = -N_2, ...N_2 - 1,$$

$$\{\sigma_{2,0}[n]\} \bigcup \{\sigma_{2,1}[n]\} \bigcup \{\sigma_{2,2}[n]\} \bigcup \{\sigma_{2,3}[n]\},\ n = -N_3, ...N_3 - 1,$$

$$\cdots$$

$$\biguplus_{l=0}^{2^M-1} \{\sigma_{M,l}[n]\},\ n = -N_{M+1}, ...N_{M+1} - 1.$$

Using the IDFT, the sequences $\sigma_{m,l}[n]$ are transformed into the sequences

$$\xi_{m,l}[k] = \frac{1}{N_m}\sum_{n=-N_m/2}^{N_m/2-1} \omega^{2^m kn}\,\sigma_{m,l}[n], \quad m = 1, ..., M,\ l = 0, ..., 2^m - 1.$$

A level–m sequence $\{\sigma_{m,l}[n]\}$ is derived by the "split" operator applied to the upper level sequence $\{\sigma_{m-1,\bar{l}}[n]\}$, $\bar{l} \stackrel{\text{def}}{=} \lfloor l/2 \rfloor$:

$$\sigma_{m,l}[n] = \mathbf{O}_{m-1}\,\sigma_{m-1,\bar{l}}[n], \quad \text{where}$$

$$\mathbf{O}_{m-1}\,\sigma_{m-1,\bar{l}}[n] \stackrel{\text{def}}{=} o_{m-1}[n]\,\sigma_{m-1,\bar{l}}[n] + o_{m-1}[n+N_m]\,\sigma_{m-1,\bar{l}}[n+N_m] \tag{9.14}$$

and the coefficients

$$o_{m-1}[n] = \begin{cases} s_{m-1}[n]/\sqrt{2}, & \text{if } \bar{l} \text{ is even } \& \ l \text{ is even or } \bar{l} \text{ is odd } \& \ l \text{ is odd;} \\ d_{m-1}[n]/\sqrt{2}, & \text{if } \bar{l} \text{ is even } \& \ l \text{ is odd or } \bar{l} \text{ is odd } \& \ l \text{ is even.} \end{cases} \tag{9.15}$$

Ultimately, the sequence $\{\sigma_{m,l}[n]\}$ is derived from the original sequence $\{\sigma[n] = \hat{x}[n]\}$ by m subsequent applications of the "split" operator:

$$\sigma_{m,l}[n] = \mathbf{O}_{m-1}\,\mathbf{O}_{m-2}\,...\mathbf{O}_0\,\sigma[n]. \tag{9.16}$$

Such a derivation scheme of $\{\sigma_{m,l}[n]\}$ determines the structure of the sequence $\{\xi_{m,l}[k]\}$, which is the IDFT of $\{\sigma_{m,l}[n]\}$. It is shown in the next section that the coefficients $\{\xi_{m,l}[k]\}$ are the inner products of the signal $\mathbf{x}$ with shifts of some signals, which are to be designated as the discrete-time spline wavelet packets. So far, it was established for the coefficients $\{\xi_{1,0}[k]\}$ $\{\xi_{1,1}[k]\}$ in the one-level transform Eq. (9.13).

9.1.3.2 Discrete-Time Wavelet Packets of Coarser Levels

Define a system of signals

$$\begin{aligned} &\Psi^{2r}_{m,l} \stackrel{\text{def}}{=} \quad \left\{\Psi^{2r}_{m,l}[k] = \tfrac{1}{N}\sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \Xi_{m,l}[n]\right\}, \\ &\quad \text{where} \\ &\Xi_{m,l}[n] \stackrel{\text{def}}{=} 2^m \prod_{\lambda=0}^{m-1} o^*_\lambda[n], \end{aligned} \tag{9.17}$$

where $o_\lambda[n]$ are defined in Eq. (9.15).

Proposition 9.2 *If a sequence* $\{\sigma_{m,l}[n]\}$, $n = -N_m/2, ..., N_m/2 - 1$, *is derived from the sequence* $\{\sigma[n] = \hat{x}[n]\}$, $n = -N/2, ..., N/2 - 1$ *by the transform* Eq. (9.16) *then its IDFT can be represented as the inner product*

$$\xi_{m,l}[k] \stackrel{\text{def}}{=} \frac{1}{N_m} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{2^m kn}\, \sigma_{m,l}[n] = \left\langle \mathbf{x},\, \Psi^{2r}_{m,l}[\cdot - 2^m k] \right\rangle. \tag{9.18}$$

Proof For all $\mu = 1, ..., m-1$, denote a partial product in Eq. (9.16) as $\tilde{\sigma}_\mu[n] = \mathbf{O}_{\mu-1}\,...\mathbf{O}_0\,\sigma[n]$. Then,

$$\sigma_{m,l}[n] = \mathbf{O}_{m-1}, \tilde{\sigma}_{m-1}[n] = o_{m-1}[n]\tilde{\sigma}_{m-1}[n] + o_{m-1}[n+N_m]\tilde{\sigma}_{m-1}[n+N_m]$$

and the IDFT

$$\xi_{m,l}[k] = \frac{1}{N_m} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{2^m kn} \, (o_{m-1}[n]\tilde{\sigma}_{m-1}[n] + o_{m-1}[n+N_m]\tilde{\sigma}_{m-1}[n+N_m])$$
$$= \frac{1}{N_m} \sum_{n=-N_{m-1}/2}^{N_{m-1}/2-1} \omega^{2^m kn} \, o_{m-1}[n]\tilde{\sigma}_{m-1}[n].$$

In turn,

$$\tilde{\sigma}_{m-1}[n] = o_{m-2}[n]\tilde{\sigma}_{m-2}[n] + o_{m-2}[n+N_{m-1}]\tilde{\sigma}_{m-2}[n+N_{m-1}]$$
$$\Longrightarrow \xi_{m,l}[k] = \frac{1}{N_m} \sum_{n=-N_{m-2}/2}^{N_{m-2}/2-1} \omega^{2^m kn} \, o_{m-1}[n] \, o_{m-2}[n]\tilde{\sigma}_{m-2}[n].$$

Iteration of this procedure leads to the relation

$$\xi_{m,l}[k] = \frac{1}{N_m} \sum_{n=-N/2}^{N/2-1} \omega^{2^m kn} \prod_{\lambda=0}^{m-1} o_\lambda[n] \, \sigma[n] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{2^m kn} \, \Xi_{m,l}[n] \, \sigma[n],$$

where $k = 0, ..., N_m - 1$. The sequence $\omega^{2^m kn} \, \Xi_{m,l}[n]$ is the DFT of the signal $\Psi^{2r}_{m,l}[\cdot - 2^m k] = \left\{\Psi^{2r}_{m,l}[\kappa - 2^m k]\right\}$, $k = 0, ..., N-1$. Then the Parseval identity implies

$$\xi_{m,l}[k] = \sum_{\kappa=0}^{N-1} \Psi^{2r}_{m,l}[\kappa - 2^m k] \, x[k] = \left\langle \mathbf{x}, \Psi^{2r}_{m,l}[\cdot - 2^m k] \right\rangle, \quad k = 0, ..., N_m - 1, \tag{9.19}$$

where $\Psi^{2r}_{m,l}$ is defined in Eq. (9.17). ■

The wavelet packet transforms of row signals or of rows of a 2D array are implemented by the MATLAB function `spl_WP_analP.m`.

Example: DFT of the second-level wavelet packets

$$\Xi_{2,0}[n] = 2s_0[n] \, s_1[n] = \nu_{2,0}[n], \quad \Xi_{2,1}[n] = 2s_0[n] \, d_1^*[n] = \nu_{2,1}[n],$$
$$\Xi_{2,2}[n] = 2d_0^*[n] \, d_1^*[n] = \nu_{2,2}[n], \quad \Xi_{2,3}[n] = 2d_0^*[n] \, s_1[n] = \nu_{2,3}[n].$$

where $n = -N/2, , N/2 - 1$.

Similarly to the discrete-time wavelets $\Psi^{2r}_{1,0}$ and $\Psi^{2r}_{1,1}$ of the first level, the DFT $\left\{\Xi_{2,l}[n]\right\}$ of the signals $\Psi^{2r}_{2,l}$, $l = 0, ..., 3$, of the second level coincide with the SHA spectra of the corresponding wavelet packets $\psi^{2r}_{2,l}(t)$, $l = 0, 1, 2, 3$. Comparing Eqs. (9.14) and (9.15) with the tree-structured scheme Eq. (8.49) for derivation of

the SHA spectra of wavelet packets, one can readily verify that this claim is true for all the levels.

Proposition 9.3 *For any level m, the DFT of the signals* $\Psi_{m,l}^{2r}$

$$\Xi_{m,l}[n] = \nu_{m,l}[n], \quad l = 0, ..., 2^{m-1}, \ n = -N/2, ..., N/2 - 1,$$

where $\{\nu_{m,l}[n]\}$ *are the SHA spectra of the respective wavelet packets* $\psi_{m,l}^{2r}(t)$.

Corollary 9.1 *The following orthogonality relations hold for the signals* $\Psi_{m,l}^{2r}$

$$\begin{aligned}
&\left\langle \Psi_{m,l}^{2r}[\cdot - 2^m\lambda], \Psi_{m,l}^{2r}[\cdot - 2^m\mu] \right\rangle \\
&= \sum_{k=0}^{N-1} \Psi_{m,l}^{2r}[k - 2^m\lambda]\, \Psi_{m,l}^{2r}[k - 2^m\mu] = \delta[\lambda - \mu] \Longrightarrow \left\| \Psi_{m,l}^{2r} \right\| = 1, \\
&\left\langle \Psi_{m,l}^{2r}[\cdot - 2^m\lambda], \Psi_{m,\tilde{l}}^{2r}[\cdot - 2^m\mu] \right\rangle = 0 \quad \textit{if } \tilde{l} \neq l.
\end{aligned}$$

The set of signals

$$\mathbf{B}_m^h \stackrel{\text{def}}{=} \bigoplus_{l=0}^{2^m-1} \left\{ \Psi_{m,l}^{2r}[k - 2^m\lambda] \right\}_{\lambda=0}^{N/2^m-1}, \tag{9.20}$$

from the space $\Pi[N]$ *forms an orthonormal basis of this space.*

The proof is similar to the proof of Proposition 9.1.

The signals $\Psi_{m,l}^{2r}$ are called the discrete-time wavelet packets of the level m. Figures 9.3, 9.4 and 9.5, produced by the MATLAB code `Di_wp_level2_P`, display the discrete-time wavelet packets of the second level, which originate from the splines of orders 2, 4 and 8, respectively. The bottom panels in the figures depict their DFT, which coincide with the SHA spectra of the corresponding spline wavelet packets. The shape of the discrete-time wavelet packets is very similar to the shape of the continuous–time wavelet packets. They are symmetric and well localized in time domain.

9.1.4 Bases of Discrete-Time Wavelet Packets

The wavelet packet transforms produce a variety of orthonormal bases for the space $\Pi[N]$. If the discrete-time wavelet packets of order $2r$ are utilized and the transform is implemented down to the level M then the following representations of a signal $\mathbf{x} \in \Pi[N]$ are possible

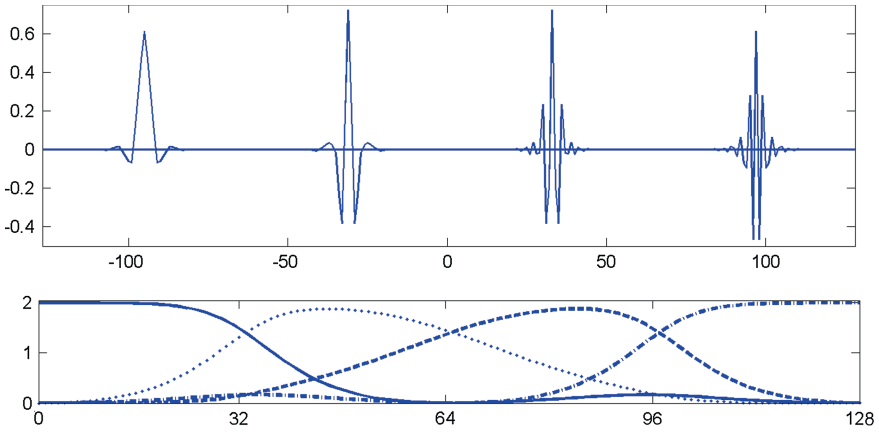


Fig. 9.3 Second-level discrete-time wavelet packets of order 2 and the right half-band of their DFT. *Upper frame* from *left* to *right* $\Psi_{2,l}^2$, $l = 0, 1, 2, 3$. *Bottom frame* $\Xi_{2,0}[n]$, (*solid line*) $\Xi_{2,1}[n]$ (*dotted*), $\Xi_{2,2}[n]$ (*dashed*), $\Xi_{2,3}[n]$ (*dashdotted*)

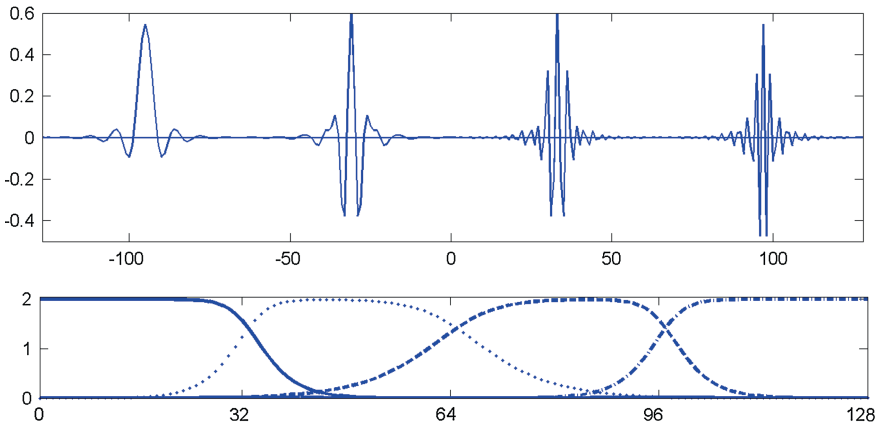


Fig. 9.4 Second-level discrete-time wavelet packets of order 4 and the right half-band of their DFT. *Upper frame* from *left* to *right* $\Psi_{2,l}^2$, $l = 0, 1, 2, 3$. *Bottom frame* $\Xi_{2,0}[n]$, (*solid line*) $\Xi_{2,1}[n]$ (*dotted*), $\Xi_{2,2}[n]$ (*dashed*), $\Xi_{2,3}[n]$ (*dashdotted*)

$$\mathbf{x} = \mathbf{x}_{1,0} \bigoplus \mathbf{x}_{1,1} = \mathbf{x}_{2,0} \bigoplus \mathbf{x}_{2,1} \bigoplus \mathbf{x}_{2,2} \bigoplus \mathbf{x}_{2,3} = \ldots = \bigoplus_{l=0}^{2^M-1} \mathbf{x}_{M,l},$$

$$\mathbf{x}_{m,l} = \left\{x_{m,l}[k]\right\}, \quad x_{m,l}[k] = \sum_{\lambda=0}^{2^m-1} \xi_{m,l}[\lambda]\, \Psi_{m,l}^{2r}[k - 2^m\lambda], \tag{9.21}$$

$$\xi_{m,l}[\lambda] = \left\langle \mathbf{x}, \Psi_{m,l}^{2r}[\cdot - 2^m\lambda] \right\rangle = \sum_{k=0}^{N-1} x[k]\, \Psi_{m,l}^{2r}[k - 2^m\lambda].$$

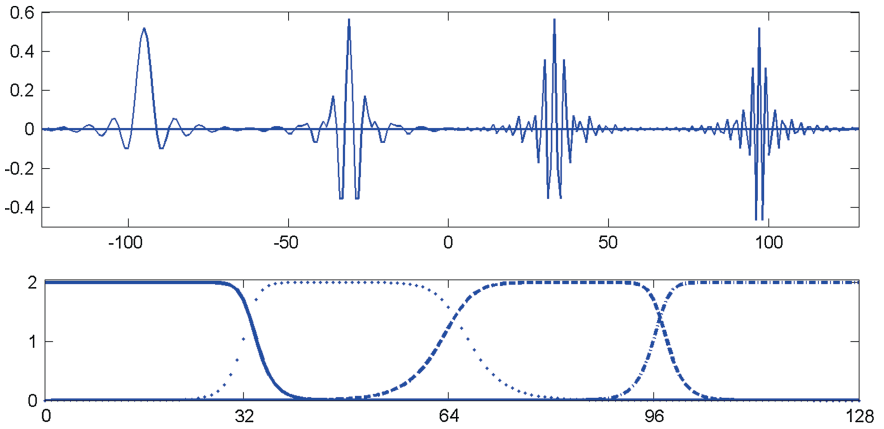


Fig. 9.5 Second-level discrete-time wavelet packets of order 8 and the right half-band of their DFT. *Upper frame* from *left* to *right* $\Psi_{2,l}^2$, $l = 0, 1, 2, 3$. *Bottom frame* $\Xi_{2,0}[n]$, (*solid line*) $\Xi_{2,1}[n]$ (*dotted*), $\Xi_{2,2}[n]$ (*dashed*), $\Xi_{2,3}[n]$ (*dashdotted*)

Respectively, the signal space becomes decomposed into a variety of mutually orthogonal subspaces.

$$\Pi[N] = \Pi[N]_{1,0} \bigoplus \Pi[N]_{1,1} = \Pi[N]_{2,0} \bigoplus \Pi[N]_{2,1} \bigoplus \Pi[N]_{2,2} \bigoplus \Pi[N]_{2,3}$$
$$= \ldots = \bigoplus_{l=0}^{2^M-1} \Pi[N]_{M,l}.$$

This decomposition is similar to the decomposition of the spline space ${}^{2r}\mathscr{S}$, which is presented by Eq. (8.24). Equation (9.14) implies that the coordinates of the signals $\mathbf{x}_{m,2l}$ and $\mathbf{x}_{m,2l+1}$ are derived from coordinates of the signal $\mathbf{x}_{m-1,l}$ from the upper level. Therefore, there exist tree-structured relations between the subspaces $\Pi[N]_{m,l}$ related to different decomposition levels, which are similar to relations between the spline subspaces ${}^{2r}\tilde{\mathscr{W}}_{m,l}$ illustrated in Fig. 8.4:

The DFT of a signal $\mathbf{x}_{m,l} \in \Pi[N]_{m,l}$, which is presented by Eq. (9.21) is determined by the DFT of the wavelet packet $\Psi_{m,l}$, which is effectively confined within the band

$$\Lambda_{m,l} \stackrel{\text{def}}{=} \left[-\frac{(l+1)N}{2^{m+1}}, -\frac{(l)N}{2^{m+1}}\right] \bigcup \left[\frac{(l)N}{2^{m+1}}, \frac{(l+1)N}{2^{m+1}}\right], \quad l = 0, \ldots, 2^m - 1. \tag{9.22}$$

Due to orthonormality of the shifts of the wavelets $\Psi_{m,l}^{2r}$, the norms of signals are:

$$\left\|\mathbf{x}_{m,l}\right\|^2 = \sum_{\lambda=0}^{N_m-1} \left|\xi_{m,l}[\lambda]\right|^2, \quad \|\mathbf{x}\|^2 = \sum_{l=0}^{2^m-1} \sum_{\lambda=0}^{N_m-1} \left|\xi_{m,l}[\lambda]\right|^2.$$

Loosely speaking, the sum $\sum_{\lambda=0}^{N_m-1} \left|\xi_{m,l}[\lambda]\right|^2 = \left\|\mathbf{x}_{m,l}\right\|^2$ characterizes the energy of the component $\mathbf{x}_{m,l} \in \Pi[N]_{m,l}$ of the signal $\mathbf{x}$, whose DFT is concentrated within the band $\Lambda_{m,l}$ defined in Eq. (9.22). Therefore, the collection $\left\{\left\|\mathbf{x}_{m,l}\right\|^2\right\}$, $l = 0, ..., 2^m - 1$, of squared norms reflects distribution of energy of the signal $\mathbf{x}$ between frequency bands $\Lambda_{m,l}$ of length $N/2^m$.

This relation between the DFT of a signal and the norms $\left\|\mathbf{x}_{m,l}\right\|^2$ is illustrated in Fig. 9.7, which was produced by the MATLAB code `bebas_examP.m`. The top frame displays right-hand part of the spectrum $\left\{\left|\hat{x}[n]\right|\right\}$, $n = 0, ..., 511$, of a signal $\mathbf{x} \in \Pi[1024]$. The subsequent frames display the values of the sums $\sum_{\lambda=0}^{N_m-1} \left|\xi_{m,l}[\lambda]\right|^2 = \left\|\mathbf{x}_{m,l}\right\|^2$, $l = 0, ..., 2^m - 1$, for the levels from $m = 6$ (second from the top frame) to $m = 1$ (bottom frame). The wavelet packet transform originating from the eight-order splines is applied.

There are 64 mutually orthogonal components of the signal $\mathbf{x} = \sum_{l=0}^{63} \mathbf{x}_{6,l}$, whose DFT effectively split the frequency band $n = -512, ..., 511$ of the signal $\mathbf{x}$ into 64 bands. The norms of the components $\mathbf{x}_{6,l}$, $l = 0, ..., 63$ characterize the contents of the signal $\mathbf{x}$ within each of the 64 frequency bands $\Lambda_{6,l}$ of width 1024/64=16. In that sense, the set of norms $\sum_{\lambda=0}^{15} \left|\xi_{6,l}[\lambda]\right|^2 = \left\|\mathbf{x}_{6,l}\right\|^2$, $l = 0, ..., 63$, can be regarded as a spectrum of frequency bands of width 16 of the signal $\mathbf{x}$. It presents an alternative to the spectrum of single frequencies, which is provided by the DFT. However, we can observe in Fig. 9.7 an apparent correlation between the DFT of the signal and the spectrum of frequency bands. The same can be said about the norms $\sum_{\lambda=0}^{31} \left|\xi_{5,l}[\lambda]\right|^2 = \left\|\mathbf{x}_{5,l}\right\|^2$, $l = 0, ..., 31$, of the signals from the fifth level, which produce the spectrum of wider frequency bands (width = 32). The signals $\mathbf{x}_{m,l}$ from upper levels produce the spectra of even wider bands.

These spectra of frequency bands appear useful in a number of signal processing applications. For example, they can serve as characteristic features of a signal or a class of signals. In order to reveal the frequency bands, which optimally represent a signal or of a class of signals, techniques, which are modifications of the Best Basis method, can be utilized. An example of such application is presented in Chap. 12. The conventional Best Basis algorithm based on the entropy cost function can be applied in order to find a representation of a signal, such that its energy is accumulated in a small number of large coefficients.

9.1.4.1 Projection of a Signal Onto $\Pi[N]_{m,l}$

Assume a signal $\mathbf{x} = \{x[k]\}$ belongs to the space $\Pi[N]$ and $\left\{\sigma[n] = \hat{x}[n]\right\}$ is its DFT. The projection of the signal onto the subspace $\Pi[N]_{m,l}$

$$x_{m,l}[k] = \sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}[\lambda]\, \Psi_{m,l}^{2r}[k - 2^m\lambda], \quad \widehat{\Psi}_{m,l}^{2r}[n] = \Xi_{m,l}^{2r}[n] = v_{m,l}[n].$$

Then

$$\Psi_{m,l}^{2r}[k-2^m\lambda] = \frac{1}{N}\sum_{n=-N/2}^{N/2} \omega^{(k-2^m\lambda)n}\, \nu_{m,l}[n]$$

and

$$\begin{aligned} x_{m,l}[k] &= \frac{1}{N}\sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}[\lambda] \sum_{n=-N/2}^{N/2} \omega^{(k-2^m\lambda)n}\, \nu_{m,l}[n] \\ &= \frac{1}{N}\sum_{n=-N/2}^{N/2} \omega^{kn}\, \nu_{m,l}[n] \sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}[\lambda]\, \omega^{-2^m\lambda n} \\ &= \frac{1}{N}\sum_{n=-N/2}^{N/2} \omega^{kn}\, \nu_{m,l}[n]\, \hat{\xi}_{m,l}[n]_m, \end{aligned}$$

where the DFT $\hat{\xi}_{m,l}[n]_m = \sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}[\lambda]\, \omega^{-2^m\lambda n}$ is an N_m-periodic sequence. Thus the projection DFT

$$\hat{x}_{m,l}[n] = \nu_{m,l}[n]\, \hat{\xi}_{m,l}[n]_m. \tag{9.23}$$

On the other hand, the projection DFT $\hat{x}_{m,l}[n]$ can be expressed via the DFT $x[n]$ of the original signal $\mathbf{x}$. This expression will be utilized in the block-based deconvolution algorithm to be presented in Chap. 11.

Proposition 9.4 *The following representation of the projection DFT holds*

$$\begin{aligned} \hat{x}_{m,l}[n] &= \nu_{m,l}[n]\, \hat{\xi}_{m,l}[n]_m \\ &= \frac{\nu_{m,l}[n]}{2^m} \sum_{\lambda=-2^m/2}^{2^m/2-1} \hat{x}[n+\lambda\, N_m]\, \nu_{m,l}^*[n+\lambda\, N_m] \approx \frac{1}{2^m}\, \hat{x}[n]\, \left|\nu_{m,l}[n]\right|^2. \end{aligned}$$

Proof For simplicity we prove this fact for the wavelet $\Psi_{1,0}^{2r}[k]$ of the first level, whose DFT is $\nu_{1,0}[n] = \sqrt{2}s_0[n]$. Proof for any other wavelet packet is quite similar.

The projected signal is $x_{1,0}[k] = \sum_{\lambda=0}^{N/2-1} \xi_{1,0}[\lambda]\, \Psi_{1,0}^{2r}[k-2\lambda]$, where the coefficient

$$\xi_{1,0}[\lambda] = \left\langle \mathbf{x}, \Psi_{1,0}^{2r}(\cdot - 2\lambda) \right\rangle = \frac{1}{N}\sum_{n=0}^{N-1} \sigma[n]\, \omega^{2\lambda n}\, \nu_{1,0}[n],$$

and $\sigma[n] = \hat{x}[n]$. Consequently, the DFT is

$$\begin{aligned}\hat{\xi}_{1,0}[n]_1 &= \sum_{\lambda=0}^{N/2-1} \omega^{-2\lambda n}\, \xi_{1,0}[\lambda] = \frac{1}{N} \sum_{k=0}^{N/2-1} \omega^{-2\lambda n} \sum_{\kappa=0}^{N-1} \sigma[\kappa]\, \omega^{2\kappa n}\, \nu_{1,0}[\kappa] \\ &= \frac{1}{N} \sum_{\kappa=0}^{N-1} \sigma[\kappa]\ \nu_{1,0}[\kappa] \sum_{\lambda=0}^{N/2-1} \omega^{2\lambda(\kappa-n)} \\ &= \frac{1}{2} \sum_{\kappa=0}^{N-1} \sigma[\kappa]\, \nu_{1,0}[\kappa]\, \delta[n-\kappa](\mathrm{mod} N/2) \\ &= \frac{1}{2} \left(\hat{x}[n]\, \nu_{1,0}[n] + \hat{x}[n+N/2]\, \nu_{1,0}[n+N/2]\right). \end{aligned} \tag{9.24}$$

Due to periodicity of the sequences, we have for $n = -N/4, ...N/4$

$$\begin{aligned}\hat{x}_{1,0}[n] &= \nu_{1,0}[n]\, \hat{\xi}_{1,0}[n]_1 = \frac{\nu_{1,0}[n]}{2} \left(\hat{x}[n]\, \nu_{1,0}[n] + \hat{x}[n+N/2]\, \nu_{1,0}[n+N/2]\right) \\ &= \frac{1}{2} \hat{x}[n]\ \left|\nu_{1,0}[n]\right|^2 + \frac{1}{2} \hat{x}[\bar{n}]\, \nu_{1,0}[n]\, \nu_{1,0}[\bar{n}], \\ \bar{n} &= \begin{cases} n + N/2, \text{ if } n < 0, \bar{n} \in (N/4, N/2), \\ n - N/2, \text{ if } n > 0, \bar{n} \in (-N/2, -N/4). \end{cases}\end{aligned}$$

Recall that the SHA spectrum $\left\{\nu_{1,0}[n]\right\}$, $n = -N/2, ..., N/2-1$ of the wavelet packet $\psi_{1,0}^{2r}(x)$ is effectively localized within the domain $[-N/4, N/4]$. The more so is true for the sequence $\left\{\left|\nu_{1,0}[n]\right|^2\right\}$, $n = -N/2, ..., N/2-1$. On the other hand, the product , $\nu_{1,0}[n]\, \nu_{1,0}[\bar{n}] \approx 0$. This leads to the relation

$$\hat{x}_{1,0}[n] \approx \frac{1}{2}\, \hat{x}[n]\ \left|\nu_{1,0}[n]\right|^2.$$

The higher is the spline's order $2r$, the closer is $\hat{x}_{1,0}[n]$ to $1/2\, \hat{x}[n]\ \left|\nu_{1,0}[n]\right|^2$ (see Fig. 8.2). ■

9.1.4.2 Best Basis

Like in the spline space ${}^{2r}\mathscr{S}$ (Sect. 8.4.4), there exists a variety of orthonormal bases of the signal space $\Pi[N]$, which are provided by different combinations of translations of discrete-time wavelet packets. Examples are the "horizontal" bases $\mathbf{B}_m^h$ defined in Eq. (9.20) or wavelet bases

$$\begin{aligned}\mathbf{B}_m^w &\stackrel{\text{def}}{=} \left\{\Psi_{m,0}^{2r}\left[\cdot - 2^m\lambda\right]\right\}_{\lambda=0}^{N_m-1} \bigcup \left\{\Psi_{m,1}^{2r}\left[\cdot - 2^m\lambda\right]\right\}_{\lambda=0}^{N_m-1} \\ &\bigcup \left\{\Psi_{m-1,1}^{2r}\left[\cdot - 2^{m-1}\lambda\right]\right\}_{\lambda=0}^{N_{m-1}-1} \bigcup ... \bigcup \left\{\Psi_{1,1}^{2r}\left[\cdot - 2\lambda\right]\right\}_{\lambda=0}^{N/2-1}.\end{aligned} \tag{9.25}$$

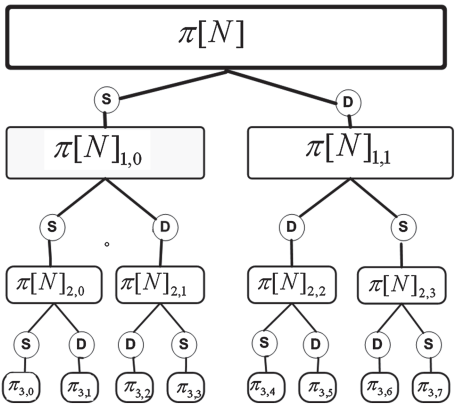


Fig. 9.6 Diagram of 3-level orthogonal decomposition of the signal space

In this case, the DFT spectrum of a signal $\mathbf{x}$ is split in a logarithmic mode.

The direct and inverse wavelet transforms of a signal are implemented by the MATLAB functions `spl_Wv_analP` and `spl_Wv_syntP`, respectively.

Besides these two kinds of orthonormal wavelet packets bases, any combination of discrete-time wavelet packets, whose DFT spectra form a one-fold cover of the whole spectral domain $n = -N/2, ...N/2 - 1$, can generate an orthonormal basis for the spline space $\Pi[N]$. Once a cost function is defined, selection of the "best" basis for a given signal from the wide variety of possible bases is implemented in line with the Best Basis scheme described in Sect. 8.4.4. This implementation is possible because the structure of the wavelet packet transforms of signals (Fig. 9.6) is identical to the tree structure of the wavelet packet transforms of splines (Fig. 8.4).

Design of the best wavelet packet basis for a signal $\mathbf{x} = \{x[k]\} \in \Pi[N]$ consists of the following steps:

1. Calculate the DFT $\sigma[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, x[k]$ of the signal $\mathbf{x}$.
2. Using the two-scale relations of Eq. (9.14), derive the coefficients

$$\sigma_{m,l}[n],\ m = 1, ..., M,\ l = 0, ..., 2^m - 1,\ n = -N_m/2, ..., N_m/2 - 1.$$

3. Applying the IDFT to the coefficients $\sigma_{m,l}[n]$, find the coordinates

$$\xi_{m,l}[k] = \frac{1}{N_m} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{2^m kn}\, \sigma_{m,l}[n],$$
$$m = 1, ..., M,\ l = 0, ..., 2^m - 1,\ k = 0, ...N_m - 1$$

of the signal $\mathbf{x}$ projections onto the subspaces $\Pi[N]_{m,l}$:

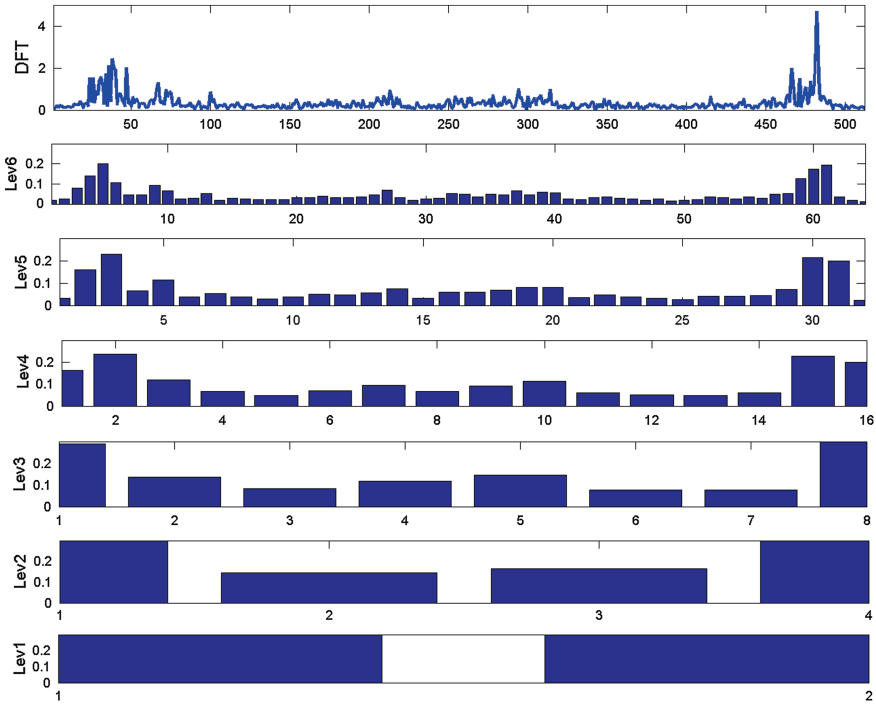


Fig. 9.7 *Top* Magnitudes of the DFT of a signal of length 1024 samples (right-hand part of the spectrum). Energies in the blocks of wavelet packet coefficients $\left\{\left\|\mathbf{x}_{m,l}\right\|^2\right\}$, $l = 0, ..., 2^m - 1$, from the sixth level (second from the *top*) down to the first level (*bottom*)

$$\mathbf{x}_{m,l} = \left\{x_{m,l}[k]\right\}, \quad x_{m,l}[k] = \sum_{\lambda=0}^{N_m-1} \xi_{m,l}[\lambda]\, \Psi_{m,l}[k - 2^m\lambda].$$

4. Calculate the cost function $C_{m,l} \stackrel{\text{def}}{=} C(\left\{\Psi_{m,l}^{2r}\left[\cdot - 2^m k\right]\right\}$ for all the bases of the subspaces $\Pi[N]_{m,l}$, $m = 1, ...M$, of the space $\Pi[N]$.
5. The best (and the single possible) basis $\bar{\mathbf{B}}^M$ for the bottom level M is the union of the discrete-time wavelet packet bases of the subspaces $\Pi[N]_{M,l}$, $l = 0, ..., 2^M - 1$.
6. The best basis $\bar{\mathbf{B}}^{M-1}$ for the upper level $M-1$ is designed in line with the rule Eq. (8.56) by comparing the cost functions of the "parent" subspaces with $\Pi[N]_{M-1,l}$ with the cost functions of their "offsprings" $\Pi[N]_{M,2l}$ and $\Pi[N]_{M,2l+1}$.
7. The procedure is iterated up to the top space $\Pi[N]$.

Once the values of the cost function are available, the best basis tree is designed by the MATLAB function `WP_BB_Listree1D_P`.

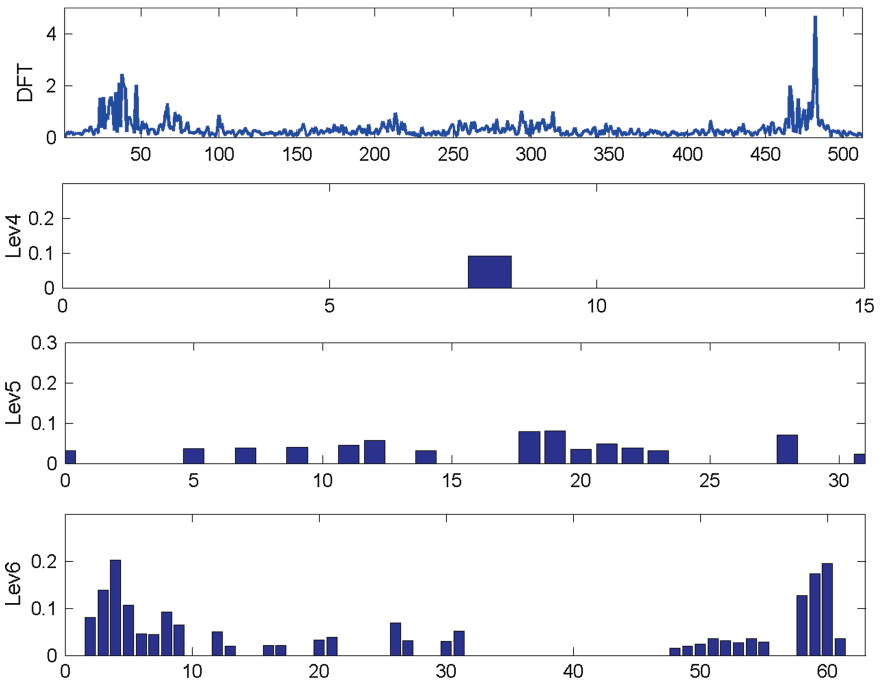


Fig. 9.8 *Top* Magnitudes of the DFT of the signal $\mathbf{x}$ of length 1024 samples, which was displayed in Fig. 9.7 (right-hand part of the spectrum). Energies in the blocks of the best-basis wavelet packet coefficients $\left\{\left\|\mathbf{x}_{m,l}\right\|^2\right\}$, $l = 0, ..., 2^m - 1$, from the fourth level (*second from the top*) down to the sixth level (*bottom*)

Example

Figure 9.8 illustrates the best basis for the signal $\mathbf{x}$, which was displayed in Fig. 9.7. The figure was produced by the same MATLAB code `bebas_examP.m` as Fig. 9.7. The best basis is formed by shifts of the discrete-time wavelet packets of eight order. The top frame displays the right-hand part of the spectrum $\left\{\left|\hat{x}[n]\right|\right\}$, $n = 0, ..., 511$, of the signal $\mathbf{x} \in \Pi[1024]$. The subsequent frames display the values of the sums of coefficients $\sum_{\lambda=0}^{N_m-1} \left|\xi_{m,l}[\lambda]\right|^2 = \left\|\mathbf{x}_{m,l}\right\|^2$ for the wavelet packets $\Psi_{m,l}^8$, which participate in the best basis.

One can observe that the wavelet packets from the decomposition levels one to three do not participate in the best basis. Only one wavelet packet $\Psi_{4,8}^8$ from the fourth decomposition level participates, while 15 wavelet packets $\Psi_{5,l}^8$, $l = 0, 7, 9, 12, 14, 18 \div 23, 28, 31$ from the fifth level and 30 wavelet packets $\Psi_{6,l}^8$ from the sixth level participate in the best basis. The DFT of the wavelet packets, whose shifts constitute the best basis, provide one-fold cover of the frequency domain of the signal $\mathbf{x}$. The list of the best-basis wavelet packets $\Psi_{m,l}^8$ and the norms $\left\|\mathbf{x}_{m,l}\right\|^2$

of the corresponding signal's projections can serve as characteristic features of the signal **x**.

Remark 9.1.2 We emphasize that design of a best discrete-time wavelet packet basis provides an optimal, with respect to the cost function, split of the frequency domain of the signal.

9.1.4.3 Reconstruction of a Signal from a Wavelet Packet Basis Expansion

Assume a **x** signal **x** is expanded over an orthonormal wavelet packet basis:

$$\mathbf{x} = \bigoplus_{(m,l)\in\mathscr{M}} \mathbf{x}_{m,l}, \quad \mathbf{x}_{m,l} = \left\{x_{m,l}[k]\right\}, \quad x_{m,l}[k] = \sum_{\lambda=0}^{N_m-1} \xi_{m,l}[\lambda]\, \Psi_{m,l}[k - 2^m\lambda]. \tag{9.26}$$

It can be a wavelet basis ($\mathscr{M} = (1, 1), ..., (M, 1), (M, 0)$), a "horizontal" basis ($\mathscr{M} = (M, 0), ..., (M, 2^M - 1)$), a best basis or some other basis. Assume that the coefficients $\left\{\xi_{m,l}[\lambda]\right\}$ are known. In order to derive an explicit expression for the component $\mathbf{x}_{m,l}$, we apply the DFT

$$\begin{aligned}
\hat{x}_{m,l}[n] &= \sum_{k=0}^{N-1} \omega^{-kn} x_{m,l}[k] = \sum_{k=0}^{N-1} \omega^{-kn} \sum_{\lambda=0}^{N_m-1} \xi_{m,l}[\lambda]\, \Psi^{2r}_{m,l}[k - 2^m\lambda] \\
&= \sum_{\lambda=0}^{N_m-1} \xi_{m,l}[\lambda] \sum_{k=0}^{N-1} \omega^{-kn}\, \Psi^{2r}_{m,l}[k - 2^m\lambda] = \sigma_{m,l}[n]\, \nu_{m,l}[n], \\
\sigma_{m,l}[n] &\stackrel{\text{def}}{=} \hat{\xi}_{m,l}[n] = \sum_{\lambda=0}^{N_m-1} \omega^{-2^m\lambda n} \xi_{m,l}[\lambda], \quad \nu_{m,l}[n] \stackrel{\text{def}}{=} \hat{\Psi}^{2r}_{m,l}[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, \Psi_{m,l}[k].
\end{aligned}$$

Then the component $\mathbf{x}_{m,l}$ is restored by the IDFT

$$x_{m,l}[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \sigma_{m,l}[n]\, \nu_{m,l}[n]$$

and the entire signal **x** is

$$x[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn} \sum_{(m,l)\in\mathscr{M}} \sigma_{m,l}[n]\, \nu_{m,l}[n]. \tag{9.27}$$

The MATLAB function `List_synthP.m` restores the signal **x** from the basis defined by a list Λ. The DFT $\nu_{m,l}[n] \stackrel{\text{def}}{=} \hat{\Psi}^{2r}_{m,l}[n]$ are prepared by the function `wavs_coefP`.

9.2 Two-Dimensional Discrete Periodic Wavelet Packets

The symbol $\Pi[N, N]$ denotes the space of N-periodic in both direction discrete signals

$$\mathbf{x} \stackrel{\text{def}}{=} \{x[k, n]\},\ k, n \in \mathbb{Z}, \quad x[k + N, n + N] = x[k, n], \quad N = 2^j.$$

In this section, a system of orthonormal bases in the space $\Pi[N, N]$ is described.

The bases to be presented are constituted by the shifts $\left\{\Psi^{2r}_{m,l,\tilde{l}}[\cdot - 2^m\lambda, \cdot - 2^m\mu]\right\}$ of the 2D discrete wavelet packets $\Psi^{2r}_{m,l,\tilde{l}}[k, n] \stackrel{\text{def}}{=} \Psi^{2r}_{m,l}[k]\,\Psi^{2r}_{m,\tilde{l}}[n]$.

9.2.1 One-Level 2D Wavelet Transform

Denote,

$$\Psi^{2r}_{1,l,\tilde{l}}[k, n] \stackrel{\text{def}}{=} \Psi^{2r}_{1,l}[k]\,\Psi^{2r}_{1,\tilde{1}}[n], \quad l, \tilde{l} = 0, 1,\ k, n = 0, ..., N - 1,$$

where the discrete-time wavelets $\Psi^{2r}_{0,1}[k] = \Psi^{2r}_{1,0}[k]$ and $\Psi^{2r}_{1,1}[k]$ are defined in Eq. (9.9). Obviously, the 2D DFT of the wavelets is equal to the SHA spectra of the 2D spline wavelet packets described in Sect. 8.5.

$$\widehat{\Psi}^{2r}_{1,l,\tilde{l}}[\kappa, \iota] = \nu_{1,l,\tilde{l}}[\kappa, \iota] \stackrel{\text{def}}{=} \nu_{1,l}[\kappa]\,\nu_{1,\tilde{l}}[\iota], \quad l, \tilde{l} = 0, 1,\ \kappa, \iota = -N/2, ..., N/2 - 1.$$

where $\nu_{1,l}[\kappa] = \widehat{\Psi}^{2r}_{1,l}[\kappa]$. Recall that

$$\nu^{2r}_{1,0}[\kappa] = \widehat{\Psi}^{2r}_{1,0}[\kappa] = \sqrt{2}\,s_0[\kappa], \quad \nu^{2r}_{1,1}[\kappa] = \sqrt{2}\,d^*_0[\kappa],$$
$$s_0[n] = \sqrt{\frac{u^{4r}[n]}{2\,u^{4r}_1[n]}}\,\cos^{2r}\frac{\pi n}{N}, \quad d_0[n] = \omega^n\,s_0[n + N/2].$$

The following fact is a direct consequence of Proposition 9.1.

Proposition 9.5 *The set of 2D wavelets*

$$\mathbf{B}^1_{2w} \stackrel{\text{def}}{=} \left\{\Psi^{2r}_{1,0,0}[k - 2\lambda, n - 2\mu]\right\} \bigcup \left\{\Psi^{2r}_{1,1,0}[k - 2\lambda, n - 2\mu]\right\}$$
$$\bigcup \left\{\Psi^{2r}_{1,0,1}[k - 2\lambda, n - 2\mu]\right\} \bigcup \left\{\Psi^{2r}_{1,1,1}[k - 2\lambda, n - 2\mu]\right\}$$
$$\lambda, \mu = 0, ..., N/2 - 1, \quad k, n = 0, ..., N - 1,$$

from the space $\Pi[N, N]$ forms an orthonormal basis of this space.

Proof The inner product of the wavelets

$$\begin{aligned}
&\left\langle \Psi^{2r}_{1,l,\tilde{l}}[k-2\lambda, n-2\mu],\ \Psi^{2r}_{1,l,\tilde{l}}[k-2\tilde{\lambda}, n-2\tilde{\mu}] \right\rangle \\
&= \sum_{k,n=0}^{N-1} \Psi^{2r}_{1,l,\tilde{l}}[k-2\lambda, n-2\mu]\, \Psi^{2r}_{1,l,\tilde{l}}[k-2\tilde{\lambda}, n-2\tilde{\mu}] \\
&= \sum_{k=0}^{N-1} \Psi^{2r}_{1,l}[k-2\lambda]\, \Psi^{2r}_{1,l}[k-2\tilde{\lambda}] \sum_{n=0}^{N-1} \Psi^{2r}_{1,\tilde{l}}[n-2\mu]\, \Psi^{2r}_{1,\tilde{l}}[n-2\tilde{\mu}] \\
&= \delta[\lambda-\tilde{\lambda}]\,\delta[\mu-\tilde{\mu}], \quad l,\tilde{l}=0,1, \quad \lambda,\mu,\tilde{\lambda},\tilde{\mu}=0,...,N/2-1.
\end{aligned}$$

Similarly, inner products of different 2D wavelets are zeros. Thus, $\mathbf{B}_2^1$ is an orthonormal system in the space $\Pi[N, N]$, which comprises N^2 elements. Consequently, $\mathbf{B}_2^1$ forms an orthonormal basis of the space $\Pi[N, N]$ ■

Any signal $\mathbf{x} \in \Pi[N, N]$ can be expanded

$$\begin{aligned}
\mathbf{x} &= \mathbf{x}_{1,0,0} \bigoplus \mathbf{x}_{1,1,0} \bigoplus \mathbf{x}_{1,0,1} \bigoplus \mathbf{x}_{1,1,1}, \\
x_{1,l,\tilde{l}}[k,n] &= \textstyle\sum_{\lambda,\mu=0}^{N/2-1} \xi_{1,l,\tilde{l}}[\lambda,\mu]\, \Psi^{2r}_{1,l,\tilde{l}}[k-2\lambda, n-2\mu],
\end{aligned} \tag{9.28}$$

where the coordinates are

$$\begin{aligned}
\xi_{1,l,\tilde{l}}[\lambda,\mu] &= \left\langle \mathbf{x},\ \Psi^{2r}_{1,l,\tilde{l}}[\cdot-2\lambda, \cdot-2\mu] \right\rangle \\
&= \sum_{k,n=0}^{N-1} x[k,n]\, \Psi^{2r}_{1,l,\tilde{l}}[k-2\lambda, n-2\mu] \\
&= \sum_{n=0}^{N-1} \Psi^{2r}_{1,\tilde{l}}[n-2\mu] \left(\sum_{k}^{N-1} x[k,n]\, \Psi^{2r}_{1,l}[k-2\lambda] \right).
\end{aligned} \tag{9.29}$$

It is seen from Eq. (9.29) that, in order to derive the coefficients $\xi_{1,l,\tilde{l}}[\lambda, \mu]$ of the 2D wavelet transform of $\mathbf{x}$ with the wavelet $\Psi^{2r}_{1,l,\tilde{l}}$, one can apply the 1D wavelet transform with the wavelet $\Psi^{2r}_{1,l}$ to columns of the array $\mathbf{x} = \{x[k, n]\}$ and, subsequently, apply the 1D wavelet transform with the wavelet $\Psi^{2r}_{1,\tilde{l}}$ to rows of the resulting array.

On the other hand, the coefficients $\xi_{1,l,\tilde{l}}[\lambda, \mu]$ can be directly computed by using the 2D IDFT:

$$\xi_{1,l,\tilde{l}}[\lambda,\mu]=\frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{2(\kappa\lambda+\iota\mu)}\sigma[\kappa,\iota]\,\nu^{*}_{1,l,\tilde{l}}[\kappa,\iota]$$
$$=\frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{2(\kappa\lambda+\iota\mu)}\sigma[\kappa,\iota]\,\nu^{*}_{1,l}[\kappa]\,\nu^{*}_{1,\tilde{l}}[\iota],\quad \sigma[\kappa,\iota]\stackrel{\text{def}}{=}\hat{x}[\kappa,\iota].$$

Inversely, if the coefficients $\xi_{1,l,\tilde{l}}[\lambda,\mu]$ are given then, similarly to Eq. (9.27), the array $\left\{x_{1,l,\tilde{l}}[k,n]\right\}$ is restored via the IDFT. Denote

$$\sigma_{1,l,\tilde{l}}[\kappa,\iota]\stackrel{\text{def}}{=}\hat{\xi}_{1,l,\tilde{l}}[\kappa,\iota]_1=\sum_{\lambda,\mu=-N/4}^{N/4-1}\omega^{-2(\kappa\lambda+\iota\mu)}\,\xi_{1,l,\tilde{l}}[\lambda,\mu].$$

Then we have

$$\begin{aligned}x_{1,l,\tilde{l}}[k,n]&=\tfrac{1}{N^2}\textstyle\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(\kappa k+\iota n)}\,\nu_{1,l,\tilde{l}}[\kappa,\iota]\,\sigma_{1,l,\tilde{l}}[\kappa,\iota]\\&=\tfrac{1}{N^2}\textstyle\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(\kappa k+\iota n)}\,\nu_{1,l}[\kappa]\,\nu_{1,\tilde{l}}[\iota]\,\sigma_{1,l,\tilde{l}}[\kappa,\iota].\end{aligned}\tag{9.30}$$

The whole signal $\mathbf{x}=\{x[k,n]\}$ is restored as

$$x[k,n]=\frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(\kappa k+\iota n)}\sum_{l,\tilde{l}=0}^{1}\nu_{1,l}[\kappa]\,\nu_{1,\tilde{l}}[\iota]\,\sigma_{1,l,\tilde{l}}[\kappa,\iota].$$

Expansion (9.28) generates decomposition of the space $\Pi[N,N]$ into four mutually orthogonal subspaces $\Pi[N,N]=\bigoplus_{l,\tilde{l}=0}^{1}\Pi_{1,l,\tilde{l}}[N,N]$, where $\Pi_{1,l,\tilde{l}}[N,N]$ is the linear hull of the wavelets

$$\left\{\Psi_{1,l,\tilde{l}}[\cdot-2\lambda,\cdot-2\mu]=\Psi_{1,l}[\cdot-2\lambda]\,\Psi_{1,\tilde{l}}[\cdot-2\mu]\right\},\quad \lambda,\mu=0,...,N/2-1.$$

9.2.2 Multilevel Wavelet Packet Transform

Let $\Psi^{2r}_{m,l},\ \Psi^{2r}_{m,\tilde{l}}\in\Pi[N],\ l,\tilde{l}=0,...2^m-1$, be discrete-time wavelet packets from level m and $\left\{\nu_{m,l}[\kappa]\right\}$, $\left\{\nu_{m,\tilde{l}}[\iota]\right\}$ be their DFT, respectively. The products

$$\Psi^{2r}_{m,l,\tilde{l}}=\left\{\Psi^{2r}_{m,l,\tilde{l}}[k,n]\right\},\quad \Psi^{2r}_{m,l,\tilde{l}}[k,n]\stackrel{\text{def}}{=}\Psi^{2r}_{m,l}[k]\,\Psi^{2r}_{m,\tilde{l}}[n]\tag{9.31}$$

are called 2D discrete wavelet packets of order $2r$ from level m. Surely, their 2D DFT

$$\left\{ v_{m,l,\tilde{l}}[\kappa, \iota] \stackrel{\text{def}}{=} \widehat{\Psi}_{m,l,\tilde{l}}[\kappa, \iota] = v_{m,l}[\kappa]\, v_{m,\tilde{l}}[\iota] \right\}, \quad \kappa, \iota = -N/2, ...N/2 - 1.$$

The following property of the 2D wavelet packets is derived from the corresponding property of the 1D wavelet packets, which is presented in Corollary 9.1.

Proposition 9.6 *The set of 2D discrete-time wavelet packets from the space $\Pi[N, N]$*

$$\mathbf{B}_m^{2h} \stackrel{\text{def}}{=} \bigcup_{l,\tilde{l}=0}^{2^m-1} \left\{ \Psi^{2r}_{m,l,\tilde{l}}[k - 2^m\lambda, n - 2^m\mu] \right\}, \tag{9.32}$$

where $\lambda, \mu = 0, ..., N/2^m - 1$, $k, n = 0, ..., N - 1$,, forms an orthonormal basis of this space.

Similarly to Eq. (9.28), a signal $\mathbf{x} \in \Pi[N, N]$ can be decomposed

$$\mathbf{x} = \bigoplus_{l,\tilde{l}=0}^{2^m-1} \mathbf{x}_{m,l,\tilde{l}}, \quad x_{m,l,\tilde{l}}[k, n] = \sum_{\lambda,\mu=0}^{N/2-1} \xi_{m,l,\tilde{l}}[\lambda, \mu]\, \Psi^{2r}_{m,l,\tilde{l}}[k - 2^m\lambda, n - 2^m\mu],$$

where the coordinates are

$$\begin{aligned} \xi_{l,l,\tilde{l}}[\lambda, \mu] &= \left\langle \mathbf{x}, \Psi^{2r}_{l,l,\tilde{l}}[\cdot - 2^m\lambda, \cdot - 2^m\mu] \right\rangle \\ &= \sum_{k,n=0}^{N-1} x[k, n]\, \Psi^{2r}_{m,l,\tilde{l}}[k - 2^m\lambda, n - 2^m\mu] \\ &= \sum_{n=0}^{N-1} \Psi^{2r}_{1,\tilde{l}}[n - 2^m\mu] \left(\sum_{k}^{N-1} x[k, n]\, \Psi^{2r}_{1,l}[k - 2^m\lambda] \right). \end{aligned} \tag{9.33}$$

It is seen from Eq. (9.33) that in order to derive the coefficients $\xi_{m,l,\tilde{l}}[\lambda, \mu]$ of the 2D wavelet transform of $\mathbf{x}$ with the wavelet $\Psi^{2r}_{1,l,\tilde{l}}$, one can apply the 1D m-level wavelet packet transform with the wavelet $\Psi^{2r}_{m,l}$ to columns of the array $\mathbf{x} = \{x[k, n]\}$ and, subsequently, apply the 1D m-level wavelet packet transform with the wavelet $\Psi^{2r}_{m,\tilde{l}}$ to rows of the resulting array. On the other hand, the coefficients $\xi_{1,l,\tilde{l}}[\lambda, \mu]$ can be directly derived by using the 2D IDFT:

$$\xi_{1,l,\tilde{l}}[\lambda,\mu] = \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{2^m(\kappa\lambda+\iota\mu)}\sigma[\kappa,\iota]\,\nu^*_{m,l,\tilde{l}}[\kappa,\iota]$$
$$= \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{2^m(\kappa\lambda+\iota\mu)}\sigma[\kappa,\iota]\,\nu^*_{m,l}[\kappa]\,\nu^*_{m,\tilde{l}}[\iota], \quad \sigma[\kappa,\iota] \stackrel{\text{def}}{=} \hat{x}[\kappa,\iota].$$

The DFT spectra $\{\nu_{m,l}[\kappa]\}$, $\left\{\nu_{m,\tilde{l}}[\iota]\right\}$ and $\left\{\nu_{m,l,\tilde{l}}[\kappa,\iota]\right\}$ of the discrete wavelet packets $\Psi_{m,l}$, $\Psi_{m,\tilde{l}}$ and $\Psi_{m,l,\tilde{l}}$ coincide with the SHA spectra of the spline wavelet packets $\psi_{m,l}(t)$, $\psi_{m,\tilde{l}}(t)$ and $\psi_{m,l,\tilde{l}}(x,y)$. Therefore, the coordinates $\left\{\xi_{m,l,\tilde{l}}[\kappa,\iota]\right\}$ of signals $\mathbf{x}_{m,l,\tilde{l}}[\kappa,\iota]$ can be derived by iterative application of the 2D wavelet packet transform, described in Sect. 8.5.2, to the signal $\mathbf{x}$. Herewith the space $\Pi[N,N]$ becomes subsequently split into sets of mutually orthogonal subspaces

$$\Pi[N,N] = \Pi_{1,0,0}[N,N]\bigoplus\Pi_{1,1,0}[N,N]\bigoplus\Pi_{1,0,1}[N,N]\bigoplus\Pi_{1,1,1}[N,N]$$
$$= \ldots\ldots\ldots\ldots$$
$$= \bigoplus_{l,\tilde{l}=0}^{2^{m-1}}\Pi_{m-1,l,\tilde{l}}[N,N] = \bigoplus_{l,\tilde{l}=0}^{2^m}\Pi_{m,l,\tilde{l}}[N,N],$$

where a subspace of level $m-1$ generates the subspaces of the next level:

$$\begin{aligned}&\Pi_{m-1,l,\tilde{l}}[N,N]\\ &= \Pi_{m,2l,2\tilde{l}}[N,N]\bigoplus\Pi_{m,2l+1,2\tilde{l}}[N,N]\bigoplus\Pi_{m,2l,2\tilde{l}+1}[N,N]\\ &\qquad\bigoplus\Pi_{m,2l+1,2\tilde{l}+1}[N,N]\end{aligned}. \tag{9.34}$$

The subspace $\Pi_{m,l,\tilde{l}}[N,N]$ is the linear hull of the wavelet packets

$$\left\{\Psi_{m,l,\tilde{l}}[\cdot-2^m\lambda,\cdot-2^m\mu] = \Psi_{m,l}[\cdot-2^m\lambda]\,\Psi_{m,\tilde{l}}[\cdot-2^m\mu]\right\}, \quad \lambda,\mu = 0,\ldots,N/2^m-1.$$

Inversely, if the coefficients $\xi_{m,l,\tilde{l}}[\lambda,\mu]$ are available then, similarly to Eq. (9.30) the array $\left\{x_{m,l,\tilde{l}}[k,n]\right\}$ is restored via the IDFT

$$x_{m,l,\tilde{l}}[k,n] = \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(\kappa k+\iota n)}\,\nu_{m,l,\tilde{l}}[\kappa,\iota]\,\sigma_{m,l,\tilde{l}}[\kappa,\iota] \tag{9.35}$$
$$= \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(\kappa k+\iota n)}\,\nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota]\,\sigma_{m,l,\tilde{l}}[\kappa,\iota],$$

where

$$\sigma_{m,l,\tilde{l}}[\kappa,\iota] \stackrel{\text{def}}{=} \hat{\xi}_{m,l,\tilde{l}}[\kappa,\iota]_m = \sum_{\lambda,\mu=0}^{N/2^m-1} \omega^{-2^m(\kappa\lambda+\iota\mu)}\, \xi_{m,l,\tilde{l}}[\lambda,\mu].$$

The whole signal $\mathbf{x} = \{x[k,n]\}$ is restored as

$$x[k,n] = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(\kappa k+\iota n)} \sum_{l,\tilde{l}=0}^{2^m-1} \nu_{m,l}[\kappa]\, \nu_{m,\tilde{l}}[\iota]\, \sigma_{m,l,\tilde{l}}[\kappa,\iota]. \tag{9.36}$$

9.2.3 2D Wavelet Bases

Equation (9.34) means that the system of orthogonal subspaces of the signal space $\Pi[N,N]$ has a quad-tree structure. The very space $\Pi[N,N]$ and each of its subspaces, except for the subspaces of lowest decomposition level, have four mutually orthogonal offsprings. Respectively, the system of orthonormal wavelet packet bases has a quad-tree structure as well. For example, an orthonormal basis of the subspace $\Pi_{m-1,l,\tilde{l}}[N,N]$ can consist either of the 2^{m-1}-sample shifts of the $(m-1)$-level discrete-time wavelet packet $\Psi^{2r}_{m-1,l,\tilde{l}}$ or, alternatively, of the 2^m-sample shifts of the quadruple $\left\{\Psi^{2r}_{m,2l,2\tilde{l}},\ \Psi^{2r}_{m,2l+1,2\tilde{l}},\ \Psi^{2r}_{m,2l,2\tilde{l}+1},\ \Psi^{2r}_{m,2l+1,2\tilde{l}+1},\ \right\}$ of the m-level wavelet packets, which are "offsprings" of the wavelet packet $\Psi^{2r}_{m-1,l,\tilde{l}}$.

9.2.3.1 2D Wavelet Bases

Thus, the wavelet packet transforms offer a vast variety of possible orthogonal expansions of a signal $\mathbf{x} \in \Pi[N,N]$. Examples are the "horizontal" orthonormal bases $\mathbf{B}^{2h}_m$ defined in Eq. (9.32) and the wavelet orthonormal bases

$$\mathbf{B}^{2w}_m \stackrel{\text{def}}{=} \left\{\Psi^{2r}_{m,0,0}[k-2^m\lambda, n-2^m\mu]\right\} \bigcup_{p=1}^{m} \Big(\left\{\Psi^{2r}_{p,1,0}[k-2^p\lambda, n-2^p\mu]\right\}$$

$$\bigcup \left\{\Psi^{2r}_{p,0,1}[k-2^p\lambda, n-2^p\mu]\right\} \bigcup \left\{\Psi^{2r}_{p,1,1}[k-2^p\lambda, n-2^p\mu]\right\} \Big).$$

Direct transform

Practically, coefficients of the signal $\mathbf{x} = \{x[k,n]\} \in \Pi[N,N]$ expansion over the wavelet basis $\mathbf{B}^{2w}_m$ are derived by application of one-level 1D wavelet transform to columns of the array $\{x[k,n]\}$, which is followed by application of one-level 1D wavelet transform to rows of the resulting array. Thus, the arrays $\left\{\xi_{1,0,0}[\lambda,\mu]\right\}$,

$\{\xi_{1,1,0}[\lambda,\mu]\}$, $\{\xi_{1,0,1}[\lambda,\mu]\}$, $\{\xi_{1,1,1}[\lambda,\mu]\}$ of size $N/2\times N/2$ are produced. Then, the same procedures are applied to the array $\{\xi_{1,0,0}[\lambda,\mu]\}$, which results in its decomposition into the arrays $\{\xi_{2,0,0}[\lambda,\mu]\}$, $\{\xi_{2,1,0}[\lambda,\mu]\}$, $\{\xi_{2,0,1}[\lambda,\mu]\}$, $\{\xi_{2,1,1}[\lambda,\mu]\}$ of size $N/4\times N/4$. Then the procedures are iterated down to level m. As a result, the set

$$\mathbf{C}_{2w}^{m}\stackrel{\text{def}}{=}\{\xi_{m,0,0}[\lambda,\mu]\}_{\lambda,\mu=0}^{N/2^m-1}\bigcup_{p=1}^{m}\big(\{\xi_{p,1,0}[\lambda,\mu]\}\bigcup\{\xi_{p,0,1}[\lambda,\mu]\}\bigcup\{\xi_{p,1,1}[\lambda,\mu]\}\big)_{\lambda,\mu=0}^{N/2^p-1}$$

of the transform coefficients, which comprises the same N^2 number of entries as the original array $\{x[k,n]\}$.

Inverse transform

Assume the set $\mathbf{C}_{2w}^{m}$ of the transform coefficients is given. Then, the signal $\mathbf{x}$ is reconstructed by the following operations:

1. The coefficients from the bottom level m are arranged into the matrix

$$\mathbf{M}_m\stackrel{\text{def}}{=}\begin{pmatrix}\{\xi_{m,0,0}[\lambda,\mu]\} & \{\xi_{m,0,1}[\lambda,\mu]\}\\ \{\xi_{m,1,0}[\lambda,\mu]\} & \{\xi_{m,1,1}[\lambda,\mu]\}\end{pmatrix},\quad \lambda,\mu=0,...,N/2^m-1.$$

2. 1D inverse wavelet transform is applied to rows of the matrix $\mathbf{M}_m$, which is followed by application of 1D inverse wavelet transform to columns of the resulting array. By this means, the array $\{\xi_{m-1,0,0}[\lambda,\mu]\}$ is restored, and used for the completion of the matrix

$$\mathbf{M}_{m-1}\stackrel{\text{def}}{=}\begin{pmatrix}\{\xi_{m-1,0,0}[\lambda,\mu]\} & \{\xi_{m-1,0,1}[\lambda,\mu]\}\\ \{\xi_{m-1,1,0}[\lambda,\mu]\} & \{\xi_{m-1,1,1}[\lambda,\mu]\}\end{pmatrix},\quad \lambda,\mu=0,...,N/2^{m-1}-1.$$

3. The restoration procedures are iterated until, at the m-step, the signal $\mathbf{x}$ is restored from the matrix

$$\mathbf{M}_1\stackrel{\text{def}}{=}\begin{pmatrix}\{\xi_{1,0,0}[\lambda,\mu]\} & \{\xi_{1,0,1}[\lambda,\mu]\}\\ \{\xi_{1,1,0}[\lambda,\mu]\} & \{\xi_{1,1,1}[\lambda,\mu]\}\end{pmatrix},\quad \lambda,\mu=0,...,N/2-1.$$

Recall that $\{\xi_{p,l,\tilde{l}}[\lambda,\mu]\}$ are the coordinates of the signal $\mathbf{x}$ projection $\mathbf{x}_{p,l,\tilde{l}}$ onto the subspace $\Pi_{p,l,\tilde{l}}[N,N]$:

$$\mathbf{x}_{p,l,\tilde{l}}[k,n]=\sum_{\lambda,\mu=0}^{,N/2^m-1}\xi_{p,l,\tilde{l}}[\lambda,\mu]\,\Psi_{p,l,\tilde{l}}^{2r}[k-2^p\lambda,n-2^p\mu].$$

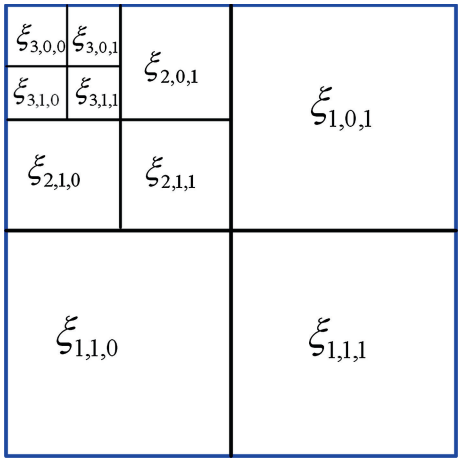


Fig. 9.9 Layout of the coefficients of 3-level 2D wavelet transform

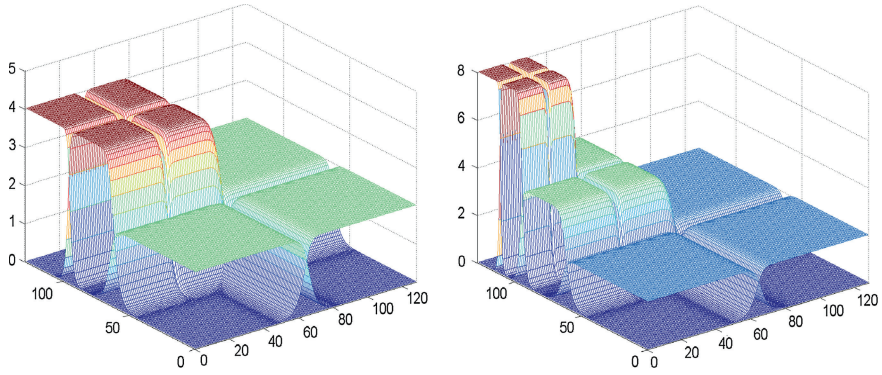


Fig. 9.10 The DFT spectra of discrete-time wavelets of order 10 down to the second (*left*) and to the third (*right*) decomposition levels

The DFT of the basis discrete-time wavelet packets $\Psi^{2r}_{p,l,\tilde{l}}$ coincide with the SHA spectra of the continuous wavelet packets $\psi^{2r}_{p,l,\tilde{l}}(t)$. The DFT of the basis 2D wavelets, which are the tensor products of the DFT of the 1F wavelets, split the 2D frequency domain of the signal $\mathbf{x}$ in a logarithmic order in either of the two directions. A conventional layout of the wavelet coefficients, which correspond with the frequency content of the respective subspaces $\Pi_{p,l,\tilde{l}}[N, N]$ is illustrated in Fig. 9.9.

The DFT of first-level wavelets are displayed in Fig. 8.19. Figure 9.10 displays the DFT of tenth order wavelets down to the second and the third decomposition levels.

The direct and inverse wavelet transforms of a 2D signal are implemented by the MATLAB functions `spl_Wv_analP_2D` and `spl_Wv_syntP_2D`, respectively.

9.2.3.2 Best basis

The variety of possible orthonormal bases, which are provided by the wavelet packet transforms, makes it possible to select a basis, which optimally represents a given signal $\mathbf{x} \in \Pi[N, N]$ or a class of signals. Fitness of a basis to the signal is measured by a cost function. The most common additive cost function is the entropy of the transform coefficients of the signal. Let a matrix $\mathbf{C} \stackrel{\text{def}}{=} \{c[k,n]\}$, $k, n = 0, ..., N-1$, comprises all the coordinates of the signal $\mathbf{x} \in \Pi[N, N]$ in some orthonormal basis $\mathbf{B}$ and $\|C\| \stackrel{\text{def}}{=} \sqrt{\sum_{k,n=0}^{N-1} |c[k,n]|^2}$. Then, the entropy cost of the basis $\mathbf{B}$ is

$$E(\mathbf{B}) \stackrel{\text{def}}{=} -\sum_{k,n=0}^{N-1} \frac{|c[k,n]|^2}{\|C\|^2} \log_e \frac{|c[k,n]|^2}{\|C\|^2}. \tag{9.37}$$

The "best" basis,

$$\mathbf{B}_2^{opt} = \bigcup_{(m,l,\tilde{l})\in\mathscr{O}} \left\{ \Psi^{2r}_{m,l,\tilde{l}}[\cdot - 2^m\lambda, \cdot - 2^m\mu] \right\}$$

which minimizes the entropy cost function for a signal $\mathbf{x} \in \Pi[N, N]$ provides a representation of the signal, where its energy is concentrated in a relatively small number of large coefficients. In addition, the best basis reveals the frequency structure of the signal.

Design of the 2D best basis is outlined in Sect. 8.5.3. Once the coordinates $\left\{\xi_{m,l,\tilde{l}}\right\}$ of the best basis expansion

$$x[k,n] = \sum_{(m,l,\tilde{l})\in\mathscr{O}} \sum_{\lambda,\mu=0}^{N/2^m-1} \xi_{m,l,\tilde{l}}\, \Psi^{2r}_{m,l,\tilde{l}}[k - 2^m\lambda, n - 2^m\mu]$$

of a signal $\mathbf{x} \in \Pi[N, N]$ are known, the signal can be reconstructed similarly to Eq. (9.36) via the IDFT

$$x[k,n] = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(\kappa k+\iota n)} \sum_{(m,l,\tilde{l})\in\mathscr{O}} \nu_{m,l}[\kappa]\, \nu_{m,\tilde{l}}[\iota]\, \sigma_{m,l,\tilde{l}}[\kappa,\iota],$$

where

$$\sigma_{m,l,\tilde{l}}[\kappa,\iota] = \hat{\xi}_{m,l,\tilde{l}}[\kappa,\iota]_m = \sum_{\lambda,\mu=0}^{N/2^m-1} \omega^{-2^m(\kappa\lambda+\iota\mu)}\, \xi_{m,l,\tilde{l}}[\lambda,\mu],$$

Fig. 9.11 "Lena" image.

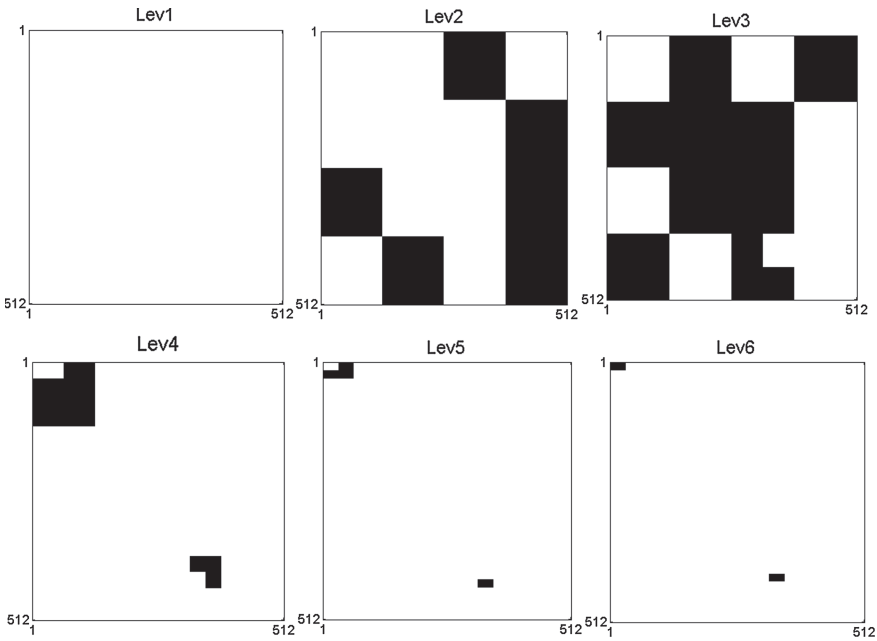


Fig. 9.12 Layout of the best basis coefficients at different levels for the six-level wavelet packet transform of the "Lena" image

$$\nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota] = \hat{\Psi}^{2r}_{m,l,\tilde{l}}[\kappa,\iota] = \sum_{\lambda,\mu=0}^{N-1} \omega^{-(\kappa\lambda+\iota\mu)}\,\Psi^{2r}_{m,l,\tilde{l}}[\lambda,\mu].$$

Another option is subsequent application of the 1D inverse transform to columns and rows of the multi-layered array of the wavelet packet coefficients provided that the coefficients blocks related to the wavelet packets, which do not participate in the best basis are put to zero. Figure 9.12, which was produced by the MATLAB code `exam_bbP_2D.m`, illustrates the layout of the best basis coefficients at different levels for the six-level wavelet packet transform of the "Lena" image (Fig. 9.11) by the fourth order wavelet packets.

Note that the layout of the blocks of the best basis coefficients for a 2D array and their relative energies highlight the structure of the array 's DFT spectrum. The set of energies in these blocks of coefficients may serve as characteristics of the array.

The list Λ of the discrete-time wavelet packets, whose shifts constitute the best basis tree is derived by the MATLAB function `WP_BB_Listree2_D_P.m`. The MATLAB function `List_synthP_2D.m` restores the 2D signal $\mathbf{x}$ from the basis defined by a list Λ and produces its DFT.

In the following Chaps. 10, 11 and 12 we discuss some applications of the spline wavelet packet analysis.

Chapter 10
Deconvolution by Regularized Matching Pursuit

Abstract In this chapter, an efficient method that restores signals from strongly noised blurred discrete data is presented. The method can be characterized as a Regularized Matching Pursuit (RMP), where dictionaries consist of spline wavelet packets. It combines ideas from spline theory, wavelet analysis and greedy algorithms. The main distinction from the conventional matching pursuit is that different dictionaries are used to test the data and to approximate the solution. In addition, oblique projections of data onto dictionary elements are used instead of orthogonal projections, which are used in the conventional Matching Pursuit (MP). The slopes of the projections and the stopping rule for the algorithm are determined automatically.

In Sect. 7.1.1 a spline algorithm for stable solution of one-dimensional convolution equations

$$g(t) = h(t) \circledast f(t) = \int_0^T h(t-\tau) f(\tau)\, d\tau, \tag{10.1}$$

which is based on Tikhonov regularization method [1, 2], is described. The unknown function $f(t)$, the blurring kernel $h(t)$ and the data function $g(t)$ are T-periodic. Utilization of splines for the Deconvolution bridges the gap between discrete noised available data and continuous nature of physical signals. The SHA methodology enables to derive a solution in an explicit form and automatically determine an optimal value of the regularization parameter.

However, spline wavelet packets provide additional opportunities for extraction of original signals from available data which is affected by blurring, sampling and corruption by noise. One of such methods is the so-called Regularized Matching Pursuit (RMP) algorithm, which is a modification of the well-known greedy Matching Pursuit (MP) algorithm [3–5]. The MP performs successive approximation of a signal by iteratively testing the available data by elements from a redundant dictionary of waveforms. The RMP uses the spline wavelet packets dictionaries and combines adaptation abilities of the MP with a regularization methodology.

Due to the great variability of shapes and frequency contents of the spline wavelet packets, they may serve as universal building blocks to approximate various functions. In addition, they are well compatible with the convolution operator. The SHA

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_10,

provides a unified computational engine for the fast implementation of the algorithm. The characteristics of the method are:

Dictionaries: RMP uses a pair of dictionaries: One, continuous, is used to approximate the solution. It consists of translations of spline wavelet packets. The other dictionary is used for testing the data. It consists of wavelet packets that are convolved with a kernel and are sampled on a grid.

Computational engine: Spline Harmonic Analysis (SHA) whose basic operation is the fast Fourier transform (FFT). It is important that the application of the FFT to the data is implemented only at the first iteration. At subsequent iterations, the correlation coefficients are updated without resorting to FFT. When the iterations are stopped, the inverse FFT is applied to obtain an explicit time domain representation of the solution.

Regularization: Orthogonal projection of data vectors onto testing dictionary elements, which are used in the MP, may result in solution instability. To avoid this, we use oblique projections, where a slope for each dictionary element is determined automatically from the data during the iterative process.

Stopping rule: Relative contributions of the coherent signal and of noise into the input data are automatically evaluated during the iterative process. When, after a number of iterations, the noise contribution becomes overwhelming, the algorithm is halted.

This approach proved to be efficient for strongly noised blurred 1D signals. Properly modified, the RMP is applied to inversion of the heat equation, which is related to the Deconvolution problem.

As in Sect. 7.1.1, the available data is being sampled on the uniform grid $\{k\}$. These samples are corrupted by noise $\mathbf{e} = \{e_k\}$, which is assumed to be a zero mean independent identically distributed (i.i.d.) random vector. Let $N = 2^j$ ($j \in \mathbb{N}$) be the number of nodes on the grid. Denote, $\mathbf{g} = \{g(k)\}_{k=0}^{N-1}$, $\mathbf{h} = \{h(k)\}_{k=0}^{N-1}$ and $\mathbf{z} = \{z[k]\} = \mathbf{g} + \mathbf{e}$. Assume that the period $T = N$.

10.1 Outline of the Matching Pursuit Scheme

10.1.1 Conventional MP

The MP algorithm [3] approximates a function by elements of a redundant dictionary of functions (features) by using a greedy strategy. The features, which best match the signal, are selected one by one from the dictionary that subsequently improves the approximation.

Let $\mathbf{D} = \{\theta_\gamma\}_{\gamma=1}^{P}$ be a dictionary with P elements and g be a given signal.

- The initial approximation is $g^0 \stackrel{\text{def}}{=} 0$ and the initial remainder is $z^0 \stackrel{\text{def}}{=} g$.

- In the first step of MP, the orthogonal projections $z^0_\gamma = \alpha_\gamma\, \theta_\gamma / \|\theta_\gamma\|$ of the remainder z^0 onto the normalized dictionary elements are calculated, where

$$\alpha_\gamma = \langle z^0, \theta_\gamma \rangle / \|\theta_\gamma\|.$$

- Assume that the projection onto the element $\theta_{\bar\gamma}$ is the strongest (that is $\bar\gamma = \arg\max_\gamma |\alpha_\gamma|$). Denote

$$\alpha^1 \stackrel{\text{def}}{=} \alpha_{\bar\gamma}, \quad \theta^1 \stackrel{\text{def}}{=} \theta_{\bar\gamma}.$$

- Define the first approximation as $g^1 = g^0 + \lambda^1 \theta^1$ and the residual as $z^1 = z^0 - \lambda^1 \theta^1$, where $\lambda^1 \stackrel{\text{def}}{=} \alpha^1 / \|\theta^1\|$.
- Then, the procedure is iterated using z^1 instead of z^0. It results in α^2, θ^2 and $\lambda^2 \stackrel{\text{def}}{=} \alpha^2 / \|\theta^2\|$.
- The second approximation is $g^2 = g^1 + \lambda^2 \theta^2$ and the residual is $z^2 = z^1 - \lambda^2 \theta^2$.
- After S iterations, we have $g = g^S + z^S = \sum_{s=0}^{S-1} \lambda^s \theta^s + z^S$.

Loosely speaking, the idea behind the MP is: The successive residuals z^s are tested for the presence of a coherent structure by elements from the dictionary **D**. If a coherent component, which is close to some dictionary element presents in the residual, then the projection of the residual onto this element is significant. Thus, this piece of the coherent structure is transferred from the residual to the approximation of the function g. If the noised version of the data is available then the residuals converge to the noise component of the data [4], which is non-coherent with all the dictionary elements. Otherwise they converge to zero [3].

10.1.2 Outline of the Regularized MP for Deconvolution

In our case, the function should be restored from its sampled convolution corrupted by noise (vector **z**). This prevents utilization of the standard MP scheme. However, we construct a pair of dictionaries and describe a modification of the MP algorithm, which enables us to accurately approximate the original function $f(t)$.

We denote, $\psi^{2r}_{m,l,k}(t) \stackrel{\text{def}}{=} \psi^{2r}_{m,l}(t+k),\ k = 0, \ldots, N-1$, where $\psi^{2r}_{m,l}(t)$ is the wavelet packet of order $2r$ from the space ${}^{2r}\mathscr{W}_{m,l}$. Let $\chi(t)$ be a spline of order $2q$ approximating the convolution kernel $h(t)$. Then, the circular convolution $\theta_{m,l}(t) \stackrel{\text{def}}{=} \chi \circledast \psi_{m,l}(t)$ is a spline of order $2q+2r$. We denote, $\theta_{m,l,k}(t) \stackrel{\text{def}}{=} \theta_{m,l}(t+k)$. Elements of the testing dictionary are defined via sampling the splines $\theta_{m,l,k}(t)$ as:

$$\Theta_{m,l} = \left\{\theta_{m,l}[\kappa]\right\}, \quad \Theta_{m,l,k} = \left\{\theta_{m,l,k}[\kappa]\right\}, \quad \kappa = 0, \ldots, N-1. \tag{10.2}$$

The algorithm is implemented via the following steps:

- Approximation of the convolution kernel $h(t)$ by a spline $\chi(t)$.

- Construction of two dictionaries:

  Approximation dictionary $\mathbf{D}_A = \{\psi^{2r}_{m,l,k}(t)\},\ k = 0, \ldots, N-1$ consisting of one-sample shifts of spline wavelet packets from different resolution scales.

  Testing dictionary $\mathbf{D}_T = \{\Theta_{m,l,k}\}$ consisting of sampled circular convolutions of shifts of spline wavelet packets with the spline $\chi(t)$ approximating the kernel $h(t)$.

- Evaluate the error vector $\mathbf{e}$: that is, construct a vector $\tilde{\mathbf{e}}$, whose characteristics are similar to the characteristics of $\mathbf{e}$.
- Calculate coefficients $\alpha_{m,l,k}$ of the *oblique* projections of $\mathbf{z}$ onto the normalized signals $\Theta_{m,l,k}$. Set a threshold T_s for the stopping rule.
- If for some triple $\check{m}, \check{l}, \check{k}$ the coefficient $|\alpha_{\check{m},\check{l},\check{k}}| < T_s$ then remove the signal $\Theta_{\check{m},\check{l},\check{k}}$ from the testing dictionary and the wavelet packet $\psi^{2r}_{\check{m},\check{l},\check{k}}(t)$ from the approximation dictionary
- Typically, the convolution kernel $h(t)$ is efficiently bandlimited. Therefore, the SHA spectra of some of remaining wavelet packets $\psi^{2r}_{\tilde{m},\tilde{l}}(t)$ actually do not overlap with the kernel spectrum. In order to avoid instability of the algorithm, discard such wavelet packets and corresponding $\Theta_{\tilde{m},\tilde{l}}$ from the dictionaries.
- Put $f^0(t) \equiv 0$, $\mathbf{z}^0 = \mathbf{z}$.
- Find *oblique* projections of $\mathbf{z}^0$ onto remaining normalized $\Theta_{m,l,k}$:

$$\mathbf{z}^0_{m,l,k} = \alpha_{m,l,k} \frac{\Theta_{m,l,k}}{\|\Theta_{m,l}\|}, \quad \alpha_{m,l,k} = \frac{\langle \Theta_{m,l,k}, \mathbf{z}^0 \rangle}{\|\Theta_{m,l}\|} \left(1 - \frac{|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}} \rangle|}{|\langle \Theta_{m,l,k}, \mathbf{z}^0 \rangle|}\right)_+ .$$

- Let $\bar{m}, \bar{l}, \bar{k} = \arg\max_{m,l,k} |\alpha_{m,l,k}|$.
- Denote $\lambda^1 \stackrel{\text{def}}{=} \alpha_{\bar{m},\bar{l},\bar{k}}/\|\Theta_{m,l}\|$, $\psi^1(t) \stackrel{\text{def}}{=} \psi^{2r}_{\bar{m},\bar{l},\bar{k}}(t)$ and $\Theta^1 \stackrel{\text{def}}{=} \Theta_{\bar{m},\bar{l},\bar{k}}$. Then, put $\mathbf{z}^1 = \mathbf{z}^0 - \lambda^1 \Theta^1$, $f^1(t) = f^0(t) + \lambda^1 \psi^1(t)$.
- The procedure is iterated using $\mathbf{z}^1$ instead of $\mathbf{z}^0$, and so on.
- The subsequent iterations of the algorithm extract pieces of coherent structure from the data signal $\mathbf{z}$. Thus, the noise-to-signal ratio tends to one and the oblique projection coefficients tend to zero. When all of them become less than a predefined threshold T_s, the iterations are automatically aborted.
- As a result, the function $f(t)$ is approximated by a finite linear combination of spline wavelet packets.

10.2 Description of the RMP for Deconvolution

The goal is to find a stable approximate solution $\tilde{f}(t)$ to the N-periodic convolution equation (Eq. (10.1)) (where $T = N$), while the data $\mathbf{z} = \{z[k]\}_{k=0}^{N-1}$ and $\mathbf{h} = \{h(k)\}_{k=0}^{N-1}$ are available, $\mathbf{z} = \mathbf{g} + \mathbf{e}$, $\mathbf{g} \stackrel{\text{def}}{=} \{g(k)\}$ and $\mathbf{e} \stackrel{\text{def}}{=} \{e_k\}_{k=0}^{N-1}$ is a random i.i.d.

zero-mean errors array. Assume that $h(t)$ and $f(t)$ are smooth functions that belong to C^p and C^s, respectively. The Fourier spectrum of the kernel h is localized in the low-frequency area: thus, the convolution with this kernel implements low-pass filtering.

10.2.1 Dictionaries for RMP

The Deconvolution problem is immersed into spline setting. The splines are represented via the orthonormal bases $\{\gamma^{2r}[n](t)\}$. The kernel $h(t) \in C^p$ is approximated by the $2q > p+1$ order spline

$$\chi(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \mu[n]\, \gamma^{2q}[n](t) \in {}^{2q}\tilde{\mathscr{S}} \subset C^p, \tag{10.3}$$

$$\mu[n] = \hat{h}[n]/V^{2q}[n], \quad V^{2q}[n] \stackrel{\text{def}}{=} \frac{u^{2q}[n]}{\sqrt{u^{4q}[n]}},$$

which interpolates the sampled kernel $\mathbf{h}$ at grid points $\{k\}$. For the successive approximation of $f \in C^s$, the spline wavelet packets of order $2r > s+1$ are used:

$$\psi^{2r}_{m,l}(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n]\, \gamma^{2r}[n](t) \in {}^{2r}\mathscr{S},$$

$m = 0, \ldots, M$, $l = 0, \ldots, 2^m - 1$, which belong to the same space C^s. The coefficients $\{\mu[n]\}$ and $\{v_{m,l}[n]\}$ form the SHA spectra of the splines $\chi(t)$ and $\psi^{2r}_{m,l}(t)$, respectively. Since the available data $\mathbf{z}$ is the sampled convolution of f with the kernel h, which was corrupted by noise, it is tested by the shifts of sampled convolution $\Theta_{m,l}[\kappa] \stackrel{\text{def}}{=} \chi \circledast \psi^{2r}_{m,l}(\kappa)$ defined in Eq. (10.2). Due to Eq. (4.33), an explicit representations of the testing signals $\{\Theta_{m,l}[\kappa]\}$ is expressed via the inverse DFT:

$$\Theta_{m,l}[\kappa] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{\kappa n}\, Q^{2r,2q}[n]\, \mu[n]\, v_{m,l}[n], \tag{10.4}$$

$$Q^{2r,2q}[n] = \frac{u^{2r+2q}[n]}{\sqrt{u^{4r}[n]\, u^{4q}[n]}}.$$

Unlike the conventional MP, the RMP uses two different dictionaries $\mathbf{D}_A$ and $\mathbf{D}_T$ for approximation of the solution and for testing the reminders, respectively. The dictionary $\mathbf{D}_A$ consists of one-sample circular shifts of the spline wavelet packets

$\psi^{2r}_{m,l}(t) \in {}^{2r}\mathscr{W}_{m,l}$, while the testing dictionary $\mathbf{D}_T$ consists of one-sample circular shifts of the signals $\Theta_{m,l}$.

The members of the dictionaries are explicitly expressed as follows:

$$\mathbf{D}_A = \left\{\psi^{2r}_{m,l,k}(t)\right\}_{k=0}^{N-1},$$

$$\psi^{2r}_{m,l,k}(t) = \psi^{2r}_{m,l}(t+k) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n]\, \omega^{kn}\, \gamma^{2r}[n](t), \tag{10.5}$$

$$\mathbf{D}_T = \left\{\Theta_{m,l,k}\right\}_{k=0}^{N-1},$$

$$\Theta_{m,l,k}(\kappa) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{(\kappa+k)n}\, Q^{2r,2q}[n]\, \mu[n]\, v_{m,l}[n], \tag{10.6}$$

where $\kappa, k = 0, \ldots, N-1$ and the sequence $Q^{2r,2q}[n]$ is defined in Eq. (10.4).

10.2.2 Modelling Noise

Assume that the error vector $\mathbf{e}$ is a zero mean Gaussian white noise. Typically, the convolution kernel $h(t)$ is efficiently bandlimited. That means its significant Fourier coefficients $c_n(h)$ occupy a relatively narrow band around zero, $-K < n < K$, where $K < N/2$. Due to Eq. (3.8), which relates the Fourier coefficients of a periodic spline to the DFT of its sampling, the same can be said about $\left\{\hat{h}[n]\right\}$. Hence, outside the band $-K < n < K$, the DFT coefficients of the data vector $\mathbf{z}$ can be expressed as

$$\left\{\hat{z}[n]\right\}_{n\in[K,N/2-1]\cup[-N/2,-K]} \approx \left\{\hat{e}[n]\right\}_{n\in[K,N/2-1]\cup[-N/2,-K]}. \tag{10.7}$$

Relying on the fact that the power spectrum, $\left\{|\hat{e}[n]|^2\right\}$ of the white noise $\mathbf{e}$ is close to a constant for all $n = -N/2, \ldots, N/2-1$, the DFT of noise is simulated by symmetric expansion of $\left\{\hat{z}[n]\right\}$ from the subset $[K, N/2-1] \cup [-N/2, -K]$ to the whole frequency domain $[-N/2+1, N/2]$. Thus, noise is modeled by the vector $\tilde{\mathbf{e}} = \{\tilde{e}_i\}_{i=0}^{N-1}$, whose DFT spectrum $\widehat{\tilde{e}}[n] = \hat{z}[n]$ as $n \in [K, N/2-1] \cup [-N/2, -K]$.

Then, the inner products $\left|\left\langle \Theta_{m,l,k},\ \mathbf{e}\right\rangle\right|$, which are needed for the oblique projections (10.11), is estimated as

$$\begin{aligned}\left|\left\langle \Theta_{m,l,k}, \mathbf{e}\right\rangle\right| &\approx e_{m,l,k} \overset{\text{def}}{=} \left|\left\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\right\rangle\right| \\ &= \frac{1}{N} \left| \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, Q^{2r,2q}[n] v_{m,l}[n]\, \mu[n]\, \widehat{\tilde{e}}[n]^* \right|.\end{aligned}$$

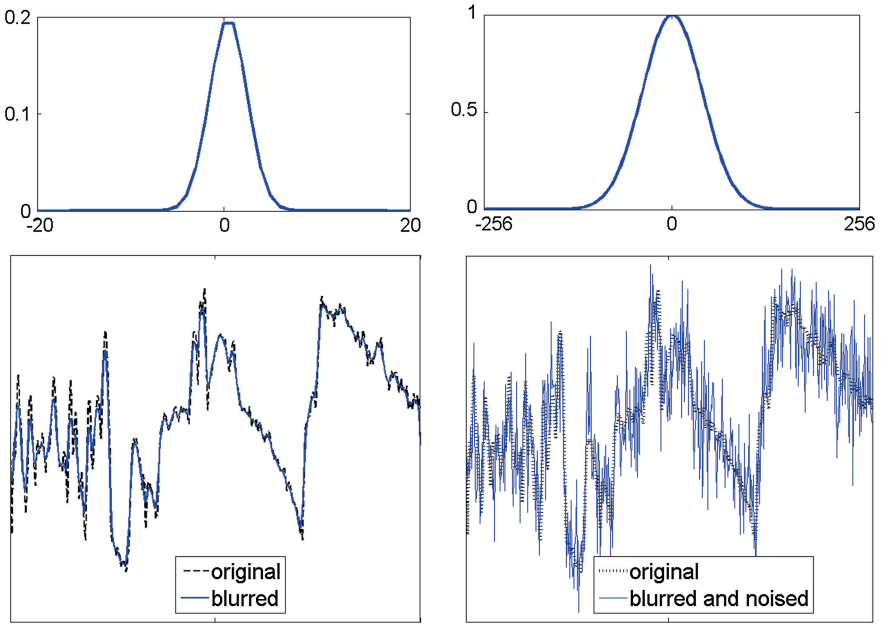


Fig. 10.1 *Top left* Blurring Gaussian kernel with STD $\sigma_k = 2$. *Top right* its magnitude Fourier spectrum (*right*). *Bottom left* column # 450 from the "Barbara" image blurred by the kernel (*solid line*) versus the original column (*dashed line*). *Bottom right* blurred column affected by the i.i.d. noise with STD $\sigma_n = 20$ (*solid line*) versus the original column (*dashed line*)

Figures 10.1 and 10.2 illustrate the idea behind noise modeling. Figure 10.1 displays the blurring kernel, its Fourier spectrum and a signal, which is the column # 450 from the blurred and noised "Barbara" image, versus the original column.

Figure 10.2 displays the magnitude DFT of the input signal versus the magnitude DFT of the applied noise. Outside the band $[-30, 30]$, both spectra are very close to each other. Outside the band $[-40, 40]$, they are practically identical. In this example, $K = 30$ (see Eq. (10.7)) is selected. The bottom frame in the figure compares the magnitude DFTs of the applied noise and the simulated one. Observe that outside the band $[-30, 30]$ both spectra are very close to each other, while inside the band their structure is similar.

10.2.3 Oblique Projection

Let the initial remainder be $\mathbf{z}^0 = \mathbf{z} = \mathbf{g} + \mathbf{e}$ and the initial approximation be $f^0(t) = 0$. The remainder is tested by the signals $\Theta_{m,l,k}$.

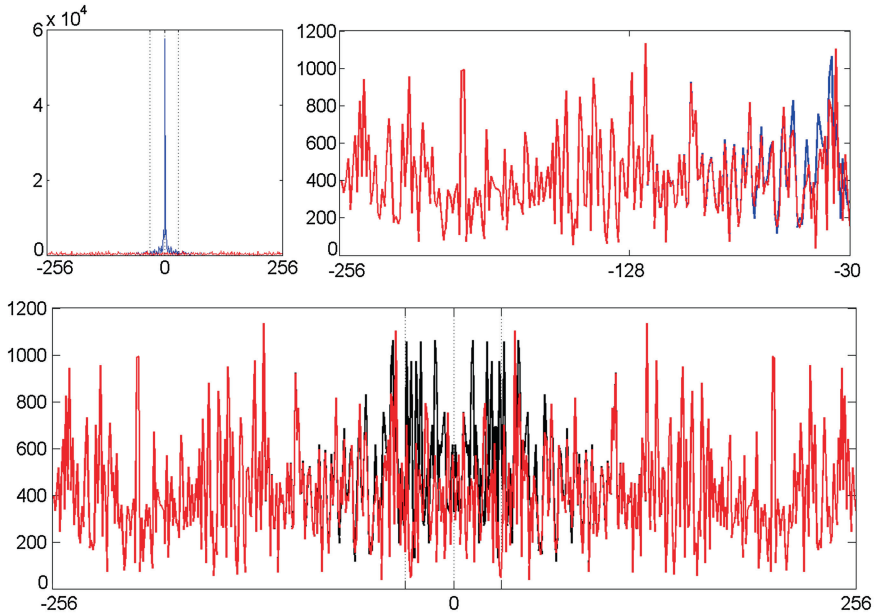


Fig. 10.2 *Top* the magnitude DFT of the column # 450 from the input image (*blue*) versus that of the applied noise (*red*). *Left* whole spectra, *right* a fragment. *Bottom* magnitude DFTs of the applied noise (*red*) versus the simulated one (*black*)

10.2.3.1 Source of Instability

In a conventional MP scheme, the search for the best-fitting dictionary element is implemented via orthogonal projection of the reminder onto the normalized dictionary elements. In our case, this approach should be modified. The orthogonal projection of $\mathbf{z}^0$ onto the normalized $\Theta_{m,l,k}$ is

$$\mathbf{z}^0_{m,l,k} = \alpha_{m,l,k}\frac{\Theta_{m,l,k}}{\|\Theta_{m,l}\|} = \lambda_{m,l,k}\Theta_{m,l,k}, \tag{10.8}$$

$$\lambda_{m,l,k} \stackrel{\text{def}}{=} \frac{\alpha_{m,l,k}}{\|\Theta_{m,l}\|} = \frac{\langle\Theta_{m,l,k}, \mathbf{z}^0\rangle}{\|\Theta_{m,l}\|^2} = \frac{\langle\Theta_{m,l,k}, \mathbf{g}\rangle}{\|\Theta_{m,l}\|^2} + \frac{\langle\Theta_{m,l,k}, \mathbf{e}\rangle}{\|\Theta_{m,l}\|^2},$$

$$\|\Theta_{m,l,k}\| = \sqrt{\sum_{i=0}^{N-1}\left(\theta_{m,l}(i+k)\right)^2} = \|\Theta_{m,l}\|.$$

The corresponding element from the approximation dictionary is $\lambda_{m,l,k}\ \psi^{2r}_{m,l,k}(t)$. If the spectral overlap of the wavelet packet $\psi^{2r}_{m,l}(t)$ with $\chi(t)$ is deficient then the norm $\|\Theta_{m,l,k}\| = \|\Theta_{m,l}\|$ of the testing element is small. Figure 10.3 displays sorted norms of the testing signals $\Theta_{m,l}$ for the cubic splines wavelet packets from the first

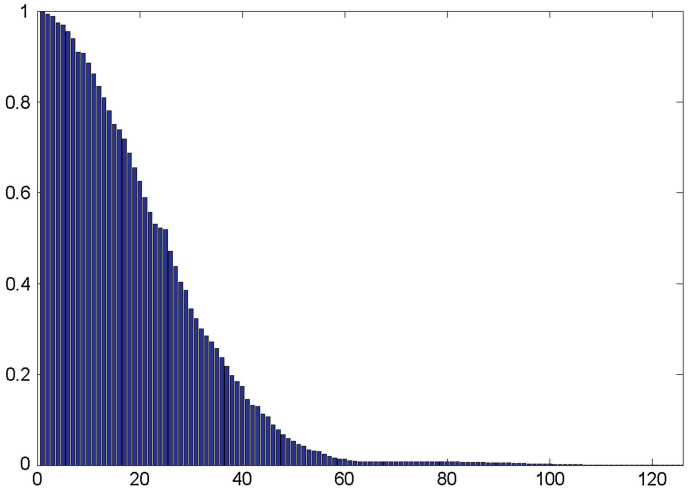


Fig. 10.3 Sorted norms of the testing signals $\Theta_{m,l}$

to sixth decomposition levels (altogether 126 wavelet packets) where the blurring kernel is displayed in Fig. 10.2. Apparently, the shape of the diagram of the sorted norms is similar to the right-hand half of the magnitude Fourier spectrum of the kernel.

Consequently, because of presence of the term $\langle \Theta_{m,l,k}, \mathbf{e} \rangle / \|\theta_{m,l}\|^2$ in Eq. (10.8), the coefficients $\lambda_{m,l,k}$ for the approximation element $\psi^{2r}_{m,l,k}(t)$ (whose norm is equal to one) can be large. This is a reason for the solution's instability. In addition, this fact hampers a correct evaluation of fitness of the elements $\psi^{2r}_{m,l,k}(t)$ for the approximation of the solution. To handle the problem, the orthogonal projection of the reminders onto the dictionary elements is replaced by an oblique projection to be described in Sect. 10.2.3.2.

10.2.3.2 Optimal Slope for Oblique Projection

Instead of the orthogonal projection coefficients $\lambda_{m,l,k} = \langle \Theta_{m,l,k}, \mathbf{z}^0 \rangle / \|\Theta_{m,l}\|^2$, the oblique projection coefficients $\grave{\lambda}_{m,l,k}$ are introduced in the following way. First, we find $\lambda_{m,l,k}(\rho)$, which minimizes the function $J(\lambda, \rho) = \rho\lambda^2 + D_{m,l,k}(\lambda, \mathbf{z}^0)$, where

$$D_{m,l,k}(\lambda, \mathbf{z}^0) \stackrel{\text{def}}{=} \sum_{\kappa=0}^{N-1} \left(\lambda \Theta_{m,l,k}(\kappa) - \lambda_{m,l,k}\ \Theta_{m,l,k}(\kappa)\right)^2$$

$$= \left(\lambda - \lambda_{m,l,k}\right)^2 \sum_{\kappa=0}^{N-1} \left(\Theta_{m,l,k}(\kappa)\right)^2 = \left(\lambda - \lambda_{m,l,k}\right)^2 \left\|\Theta_{m,l}\right\|^2,$$

and ρ is a numerical parameter. The minimum is achieved when

$$\lambda = \lambda_{m,l,k}(\rho) = \frac{\lambda_{m,l,k} \left\| \theta_{m,l} \right\|^2}{\rho + \left\| \Theta_{m,l} \right\|^2} = \frac{\left\langle \Theta_{m,l,k}, \mathbf{z}^0 \right\rangle}{\rho + \left\| \Theta_{m,l} \right\|^2}. \tag{10.9}$$

The function $\lambda_{m,l,k}(\rho)$ differs from the orthogonal projection coefficient $\lambda_{m,l,k}$ by the addend ρ in the denominator. The smaller the parameter ρ is, the closer $\lambda_{m,l,k}(\rho)$ is to $\lambda_{m,l,k}$. Denote,

$$d_{m,l,k}(\rho) \stackrel{\text{def}}{=} D_{m,l,k}(\lambda_{m,l,k}(\rho), \mathbf{z}^0) = \left(\frac{Z_{m,l,k}\ \rho}{\rho + \left\| \Theta_{m,l} \right\|^2} \right)^2,$$

$$\text{where } Z_{m,l,k} \stackrel{\text{def}}{=} \frac{\left| \left\langle \Theta_{m,l,k}, \mathbf{z}^0 \right\rangle \right|}{\left\| \Theta_{m,l} \right\|}$$

is the norm of the orthogonal projection of $\mathbf{z}^0$ onto $\Theta_{m,l,k}/\|\Theta_{m,l,k}\| = \Theta_{m,l,k}/\|\Theta_{m,l}\|$. The function $\lambda_{m,l,k}(\rho)$ decays monotonically from $\lambda_{m,l,k}(0) = \lambda_{m,l,k}$ to $\lambda_{m,l,k}(\infty) = 0$. Once the value of the parameter ρ is properly selected, $\lambda_{m,l,k}(\rho)$ is used as the coefficient for the element $\psi_{m,l}^{2r}(t)$, which approximates the solution. For this approximation to be stable, this coefficient should not be too large even if the norm $\|\Theta_{m,l}\|^2$ is small. This is achieved by increasing ρ. On the other hand, the function $d_{m,l,k}(\rho)$, which is the squared norm of deviation of the oblique $\lambda_{m,l,k}(\rho)\Theta_{m,l,k}$ from the orthogonal $\lambda_{m,l,k}\Theta_{m,l,k}$ projection of $\mathbf{z}^0$ onto the testing element $\Theta_{m,l,k}$, grows monotonically from $d_{m,l,k}(0) = 0$ to $d_{m,l,k}(\infty) = \left(Z_{m,l,k}\right)^2$.

It does not make sense for the oblique projection to approach the orthogonal projection closer than the noise level. Therefore, we can increase ρ (thus reducing $\lambda_{m,l,k}(\rho)$) until

$$d_{m,l,k}(\rho) = \left(\frac{Z_{m,l,k}\ \rho}{\rho + \left\| \Theta_{m,l} \right\|^2} \right)^2 = \left(E_{m,l,k}\right)^2, \quad \text{where} \tag{10.10}$$

$$E_{m,l,k} \stackrel{\text{def}}{=} \frac{\left| \left\langle \Theta_{m,l,k}, \tilde{\mathbf{e}} \right\rangle \right|}{\left\| \Theta_{m,l} \right\|}$$

is the orthogonal projection norm of the evaluated error vector $\tilde{\mathbf{e}}$ onto $\Theta_{m,l,k}/\left\|\Theta_{m,l}\right\|$.

The parameter ρ derived from Eq. (10.10) is

$$\rho_{m,l,k} = \frac{E_{m,l,k} \left\| \Theta_{m,l} \right\|^2}{\left(Z_{m,l,k} - E_{m,l,k}\right)}.$$

Hence, an optimal slope is

$$\grave{\lambda}_{m,l,k} = \lambda_{m,l,k}\left(1 - \frac{E_{m,l,k}}{Z_{m,l.k}}\right) = \frac{\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle}{\|\Theta_{m,l}\|^2}\left(1 - \frac{|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle|}{|\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle|}\right)$$

and the oblique projection of $\mathbf{z}^0$ onto the normalized $\Theta_{m,l,k}$ is

$$\mathbf{z}^0_{m,l,k} = \grave{\alpha}_{m,l,k}\frac{\Theta_{m,l,k}}{\|\Theta_{m,l}\|}, \quad \text{where} \tag{10.11}$$

$$\grave{\alpha}_{m,l,k} = \|\Theta_{m,l}\|\grave{\lambda}_{m,l,k} = \frac{\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle}{\|\Theta_{m,l}\|}\left(1 - \frac{|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle|}{|\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle|}\right)$$

can be referred to as the "oblique inner product" of the remainder $\mathbf{z}^0$ with the normalized testing signal $\Theta_{m,l,k}$.

Remark 10.2.1 Due to the fact that the estimated $\tilde{\mathbf{e}}$ rather than the real $\mathbf{e}$ noise vector involved, it may happen that, for some $\Theta_{m,l,k}$, the value of $|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle|$ exceeds $|\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle|$. This means that within the time-frequency range of the testing signal $\Theta_{m,l,k}$ estimated noise $\tilde{\mathbf{e}}$ prevails over the signal $\mathbf{z}^0$. Apparently, contribution of the respective wavelet packet $\psi^{2r}_{m,l,k}(t)$ will be erroneous. Therefore, the oblique projection coefficients are modified by including the "control" function $x_+ \stackrel{\text{def}}{=} (x + |x|)/2$. Thus, the modified coefficients are

$$\acute{\lambda}_{m,l,k} = \lambda_{m,l,k}\left(1 - \frac{E_{m,l,k}}{Z_{m,l.k}}\right)_+, \quad \acute{\alpha}_{m,l,k} = \|\Theta_{m,l}\|\acute{\lambda}_{m,l,k}. \tag{10.12}$$

10.2.4 Stopping Threshold and Reduction of Dictionaries

Let the initial remainder be $\mathbf{z}^0 = \mathbf{z} = \mathbf{g} + \mathbf{e}$ and the estimated noise vector be $\tilde{\mathbf{e}}$. The remainder is tested by multiplication of $\mathbf{z}^0$ with the normalized signals $\Theta_{m,l,k}$. The "oblique inner products" of the remainder $\mathbf{z}^0$ with the normalized testing signal $\Theta_{m,l,k}$ are

$$\acute{\alpha}_{m,l,k} = \alpha_{m,l,k}\left(1 - \frac{|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle|}{|\langle \Theta_{m,l,k}, \mathbf{z}^0\rangle|}\right)_+,$$

where the inner product

$$\alpha_{m,l,k} \stackrel{\text{def}}{=} \frac{\langle \Theta_{m,l,k}, \mathbf{z}^0 \rangle}{\|\Theta_{m,l}\|} = \alpha^{\mathbf{g}}_{m,l,k} + \alpha^{\mathbf{e}}_{m,l,k}$$

$$\alpha^{\mathbf{g}}_{m,l,k} \stackrel{\text{def}}{=} \frac{\langle \Theta_{m,l,k}, \mathbf{g} \rangle}{\|\Theta_{m,l}\|}, \quad \alpha^{\mathbf{e}}_{m,l,k} \stackrel{\text{def}}{=} \frac{\langle \Theta_{m,l,k}, \mathbf{e} \rangle}{\|\Theta_{m,l}\|}$$

consists of two parts: one stems from the structured signal $\mathbf{g}$ while the other originates from the random i.i.d. zero-mean errors array $\mathbf{e}$. Some of the selected testing signals $\Theta_{m,l,k}$ correlate well with the signal $\mathbf{g}$. That means that the term $\alpha^{\mathbf{g}}_{m,l,k}$ is significant. However, a number of the signals $\Theta_{\bar{m},\bar{l},\bar{k}}$ practically do not correlate with $\mathbf{g}$. For such signals $\alpha_{\bar{m},\bar{l},\bar{k}} \approx \alpha^{\mathbf{e}}_{\bar{m},\bar{l},\bar{k}}$. Therefore, the fraction

$$\frac{\left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \mathbf{e} \rangle\right|}{\left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \mathbf{z}^0 \rangle\right|} = \frac{\alpha^{\mathbf{e}}_{\bar{m},\bar{l},\bar{k}}}{\alpha_{\bar{m},\bar{l},\bar{k}}}$$

is close to one. To some extent, this claim is true for the fraction $\left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \tilde{\mathbf{e}} \rangle\right| / \left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \mathbf{z}^0 \rangle\right|$. Thus, for such signals $\Theta_{\bar{m},\bar{l},\bar{k}}$, the factor $1 - \left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \tilde{\mathbf{e}} \rangle\right| / \left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \mathbf{z}^0 \rangle\right|$ is small.

Moreover. the i.i.d. errors array $\mathbf{e}$ correlates with none of the test signals. Therefore, the inner products coefficients $\alpha^{\mathbf{e}}_{m,l,k}$ are relatively small for all the sets of $\{m, l, k\}$. Consequently,

$$\acute{\alpha}_{\bar{m},\bar{l},\bar{k}} \approx \alpha^{\mathbf{e}}_{\bar{m},\bar{l},\bar{k}} \left(1 - \frac{\left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \tilde{\mathbf{e}} \rangle\right|}{\left|\langle \Theta_{\bar{m},\bar{l},\bar{k}}, \mathbf{e} \rangle\right|}\right)_+$$

is close, if not equal, to zero. To remove such signals from the dictionaries, the values $A_{m,l} \stackrel{\text{def}}{=} \max_k |\acute{\alpha}_{m,l,k}|$, $k = 0, \ldots, N-1$, are computed for all the sets $m = 1, \ldots, M$, $l = 0, \ldots 2^m - 1$, and a threshold T_s is defined. The signals $\Theta_{\bar{m},\bar{l},\bar{k}}$ such that $A_{\bar{m},\bar{l}} < T_s$ are discarded from the testing dictionary $\mathbf{D}_T$. The respective wavelet packets $\psi^{2r}_{\bar{m},\bar{l},\bar{k}}$ are discarded from the approximation dictionary $\mathbf{D}_A$.

It is explained in Sect. 10.2.5 how setting the threshold T_s provides a stopping rule for pursuit iterations.

Due to the fact that the convolution kernel $h(t)$ is bandlimited, the norms $\|\Theta_{m,l,k}\| = \|\Theta_{m,l}\|$ of a number of the testing elements are small (see Fig. 10.3). It may happen that some of the signals $\Theta_{m,l}$, which have survived the thresholding process, have small norms. In order to avoid solution instability, such signals are removed from the already reduced dictionary.

Figure 10.4 illustrates the dictionaries reduction procedures. The top left panel displays the sorted values $A_{m,l} \stackrel{\text{def}}{=} \max_k |\acute{\alpha}_{m,l,k}|$, $k = 0, \ldots, N-1$, where $m = 1, \ldots, 6$, $l = 0, \ldots 2^m - 1$. Altogether, there are 126 testing signals $\Theta_{m,l}$. Their magnitudes drop sharply. The top right panel presents a close-up view of the smallest values $A_{m,l}$ starting from #42. This close-up view suggests that $T_s = 1$ can be a proper

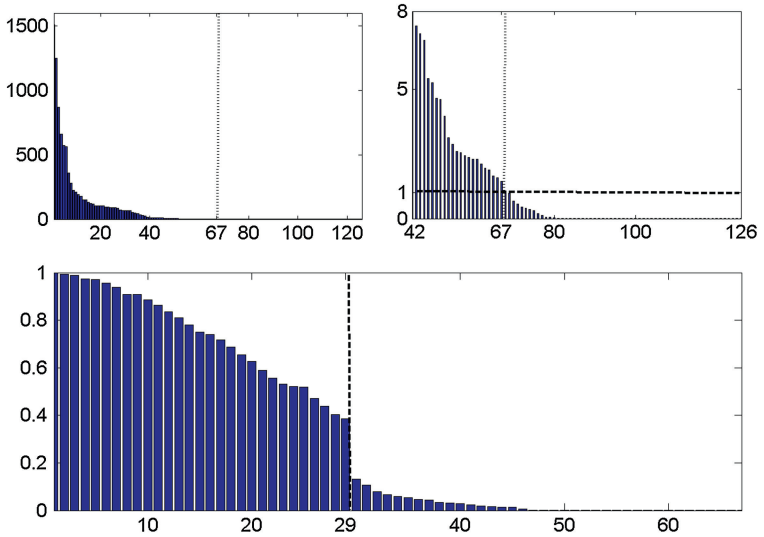


Fig. 10.4 *Top left* sorted values $A_{m,l}$. *Top right* close-up view of the smallest values $A_{m,l}$. *Bottom* sorted norms of 67 testing signals $\Theta_{m,l}$

threshold. This choice reduces the number of signals $\Theta_{m,l}$ to 67. Their sorted norms are displayed in the bottom panel of the figure. It is apparent that only the strongest 29 signals should be retained.

10.2.5 Pursuit

The dictionary element $\Theta_{m,l,k}$, which fits best to the signal $\mathbf{z}^0$, is found from comparison of magnitudes of the oblique projection coefficients $\acute{\alpha}_{m,l,k}$. Let $(\tilde{m}, \tilde{l}, \tilde{k}) = \arg\max_{m,l,k} \left|\acute{\alpha}_{m,l,k}\right|$. We denote,

$$\lambda^1 \stackrel{\text{def}}{=} \acute{\lambda}_{\tilde{m},\tilde{l},\tilde{k}} = \acute{\alpha}_{\tilde{m},\tilde{l},\tilde{k}} / \left\| \Theta_{\tilde{m},\tilde{l}} \right\|, \quad \psi^1(t) \stackrel{\text{def}}{=} \psi^{2r}_{\tilde{m},\tilde{l},\tilde{k}}(t), \quad \Theta^1 \stackrel{\text{def}}{=} \Theta_{\tilde{m},\tilde{l},\tilde{k}},$$

and put

$$\mathbf{z}^1 = \mathbf{z}^0 - \lambda^1 \Theta^1, \quad f^1(t) = f^0(t) + \lambda^1 \psi^1(t).$$

Then, the procedure is iterated using $\mathbf{z}^1$ instead of $\mathbf{z}^0$.

It is seen from Eq. (10.11) that the oblique projection coefficients $\acute{\alpha}_{m,l,k}$ depend on relative shares of the coherent structure and of noise in the projection of the remainder onto the normalized testing signal $\Theta_{m,l,k}$. The bigger is the noise share, the smaller the coefficients are. Subsequent pursuit steps extract pieces of coherent structure from the signal $\mathbf{z}$. Therefore, the noise share in the reminder increases

and the oblique projection coefficients tend to zero. Thus, one after another, they appear below the threshold T_s. Consequently, one after another signal $\Theta_{m,l}$ leaves the testing dictionary. Finally, after some K iterations, the dictionary becomes empty thus terminating the iteration process. The approximate solution is

$$\tilde{f}(t) = f^K(t) = \sum_{k=1}^{K} \lambda^k \psi^k(t). \tag{10.13}$$

10.2.6 Computational Scheme

The SHA provides efficient tools for fast implementation of the RMP algorithm. When we have to restore a signal or a set of signals, which are blurred by similar kernels and corrupted by white noise with the similar characteristics, some preliminary operations should be conducted.

10.2.6.1 Preprocessing

Dictionaries: We choose order $2q$ of the spline $\chi(t)$, which interpolates the sampled kernel $\mathbf{h} = \{h(k)\}$, $k = 0, \ldots N-1$. Its SHA coordinates are calculated explicitly, using the fast Fourier transform (FFT)

$$\chi(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \mu[n]\, \gamma^{2q}[n](t) \in {}^{2q}\tilde{\mathscr{S}} \subset C^p, \quad \mu[n] = \frac{\hat{h}[n]}{V^{2q}[n]},$$

the sequence $V^{2q}[n]$ having been defined in Eq. (10.3).

Basing on a prior information on the smoothness of the solution $f(t) \in C^s$, choose the depth M and the order $2r \geq s+1$ of the spline wavelet packets, which are used for the solution approximation, and calculate their SHA spectra $\{v_{m,l}[n]\}$, using Eq. (8.49):

$$\psi^{2r}_{m,l}(t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n]\, \gamma^{2r}[n](t) \in {}^{2r}\mathscr{S} \subset C^s,$$

where $r = 0, \ldots, M$, $l = 0, \ldots, 2^m - 1$. Then, the elements $\psi^{2r}_{m,l,k}(t)$ of the approximation dictionary $\mathbf{D}_A$ are given by Eq. (10.5).

From the coordinates $\mu[n]$ and $v_{m,l}[n]$, derive the elements

$$\Theta_{m,l,k} = \left\{ \Theta_{m,l,k}(\kappa) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{(k+\kappa)n}\, Q^{2r,2q}[n]\, v_{m,l}[n]\, \mu[n] \right\}_{\kappa=0}^{N-1},$$

$$Q^{2r,2q}[n] = \frac{u^{2r+2q}[n]}{\sqrt{u^{4r}[n]\,u^{4q}[n]}} \quad k = 0, \ldots, N-1,$$

of the testing dictionary $\mathbf{D}_T$. For all $k = 0, \ldots, N-1$, the norms are

$$\left\|\Theta_{m,l,k}\right\| = \left\|\Theta_{m,l}\right\| = \sqrt{\sum_{i=0}^{N-1} \Theta_{m,l}(i)^2} = \sqrt{\frac{1}{N}\sum_{n=-N/2}^{N/2-1} \left|Q^{2r,2q}[n]\, v_{m,l}[n]\, \mu[n]\right|^2}.$$

Modeling the noise: Additive i.i.d. noise $\mathbf{e}$ is estimated as described in Sect. 10.2.2. The vector $\tilde{\mathbf{e}}$ is the estimation of $\mathbf{e}$ and $\left\{\widehat{\tilde{e}}[n]\right\}$ is its DFT.

Setting threshold and reduction of dictionaries: Calculate the inner products $\left|\langle \Theta_{m,l,k}, \mathbf{z}\rangle\right|$ and $\left|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle\right|$, which are needed for the oblique projections (10.11), as

$$T_{m,l,k}(\mathbf{z}) \stackrel{\text{def}}{=} \left|\langle \Theta_{m,l,k}, \mathbf{z}\rangle\right| = \frac{1}{N}\left|\sum_{n=-N/2}^{N/2-1} \omega^{nk}\, Q^{2r,2q}[n] v_{m,l}[n]\, \mu[n]\, \hat{z}[n]^*\right|,$$

$$e_{m,l,k} \stackrel{\text{def}}{=} \left|\langle \Theta_{m,l,k}, \tilde{\mathbf{e}}\rangle\right| = \frac{1}{N}\left|\sum_{n=-N/2}^{N/2-1} \omega^{nk}\, Q^{2r,2q}[n] v_{m,l}[n]\, \mu[n]\, \widehat{\tilde{e}}[n]^*\right|.$$

Calculate the oblique projection coefficients

$$\acute{\alpha}_{m,l,k}(\mathbf{z}) = \frac{T_{m,l,k}(\mathbf{z})}{\left\|\Theta_{m,l}\right\|}\left(1 - \frac{e_{m,l,k}}{\left|T_{m,l,k}(\mathbf{z})\right|}\right)_+$$

and the values $A_{m,l}(\mathbf{z}) \stackrel{\text{def}}{=} \max|\acute{\alpha}_{m,l,k}(\mathbf{z})|$ with respect to $k = 0, \ldots, N-1$.

Then, as described in Sect. 10.2.4, we set the threshold T_s and reduce dictionaries $\mathbf{D}_T$ and $\mathbf{D}_A$.

Initialization of the algorithm: We approximate the solution $f(t)$ of Eq. (10.1) by a sequence of splines

$$f^j(t) = \frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1} \sigma^j[n]\, \gamma^{2r}[n](t) \in {}^{2r}\mathscr{S},$$

which are combinations of the wavelet packets from the reduced dictionary $\tilde{\mathbf{D}}_A$.

Define the initial approximation $f^0(t) \equiv 0$ and the initial remainder $\mathbf{z}^0(t) \stackrel{\text{def}}{=} \mathbf{z}$. Then, we have

$$T_{m,l,k}(\mathbf{z}^0) = T_{m,l,k}(\mathbf{z}), \quad \acute{\alpha}_{m,l,k}(\mathbf{z}^0) = \acute{\alpha}_{m,l,k}(\mathbf{z}), \quad \sigma^0[n] = 0.$$

Finally, put $j = 0$.

10.2.6.2 Pursuit

1. Find $\tilde{m}, \tilde{l}, \tilde{k} = \arg\max_{m,l,\lambda} \left|\acute{\alpha}_{m,l,k}(\mathbf{z}^j)\right|$.
2. Denote $\lambda^{j+1} \stackrel{\text{def}}{=} \acute{\alpha}_{\tilde{m},\tilde{l},\tilde{k}} / \left\|\Theta_{\tilde{m},\tilde{l}}\right\|$.
3. Put $\sigma^{j+1}[n] = \sigma^j[n] + \lambda^{j+1}\, v_{\tilde{m},\tilde{l}}[n]\, \omega^{\tilde{k}n},\ n = -N/2, \ldots, N/2-1$.
4. Put $\hat{z}^{j+1}[n] = \hat{z}^j[n] - \lambda^{j+1}\, Q^{2r,2q}[n]\, \mu[n]\, v_{\tilde{m},\tilde{l}}[n]\, \omega^{\tilde{k}n},\ n = -N/2, \ldots, N/2-1$.
5. Calculate the inner products

$$T_{m,l,k}(\mathbf{z}^{j+1}) = \frac{1}{N} \left| \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, Q^{2r,2q}[n]\, v_{m,l}[n]\, \mu[n]\, \hat{z}^{j+1}[n]^* \right|.$$

6. Calculate the oblique projection coefficients

$$\acute{\alpha}_{m,l,k}(\mathbf{z}^{j+1}) = \frac{T_{m,l,k}(\mathbf{z}^{j+1})}{\left\|\Theta_{m,l}\right\|} \left(1 - \frac{e_{m,l,k}}{\left|T_{m,l,k}(\mathbf{z}^{j+1})\right|}\right)_+$$

   and $A_{m,l}(\mathbf{z}^{j+1}) \stackrel{\text{def}}{=} \max |\acute{\alpha}_{m,l,k}(\mathbf{z}^{j+1})|$ with respect to $k = 0, \ldots, N-1$.
7. If for some $(\bar{m}, \bar{l})$ the value $A_{\bar{m},\bar{l}}(\mathbf{z}^{j+1}) < T_s$ then remove $\psi^{2r}_{\bar{m},\bar{l},k}(t)$ and $\Theta_{\bar{m},\bar{l},k},\ k = 0, \ldots N-1$, from the dictionaries $\mathbf{D}_A$ and $\mathbf{D}_T$, respectively.

If the remaining dictionaries are not empty $j = j+1$, go to 1. Otherwise: the solution is

$$f(t) \approx f^j(t) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma^j[n]\, \gamma^{2r}[n](t).$$

Calculate the grid points samples of the approximate solution by IFFT, using Eq. (4.30):

$$f^j(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\, \sigma^j[n].$$

If need be, calculate the values of the spline $f^j(t)$ at dyadic or triadic rational points by the subdivision schemes described in Sects. 6.2 and 6.3.

Remark 10.1 The described algorithm can be utilized for restoration of signals, which are corrupted by noise (no blurring). In this case, the testing signals $\{\Theta_{m,l,k}\}$ in the dictionary $\mathbf{D}_T$ should be replaced by the sampled wavelet packets

$$\mathbf{D}_T = \left\{\psi_{m,l,k}\right\}_{k=0}^{N-1}, \quad \psi_{m,l,k} = \left\{\psi_{m,l}^{2r}(\kappa + k)\right\} \tag{10.14}$$

$$\psi_{m,l}^{2r}(\kappa + k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{(k+\kappa)n} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} v_{m,l}[n], \quad k = 0, \ldots, N-1.$$

The above algorithm is implemented by the MATLAB code `RMP_runP.m` in an interactive way.

10.3 RMP for Inversion of Heat Equation

In this section we briefly outline application of the RMP method for the inversion of the heat equation

$$\frac{\partial g(x,t)}{\partial t} = g_x''(x,t),\ g(x,0) = f(x), \tag{10.15}$$

where $f(x)$ is assumed to be N-periodic.

As in Sect. 7.2.1, where a spline algorithm based on the Tikhonov regularization is described, $\mathbf{U}_t$ denotes the linear operator such that $\mathbf{U}_t f(x) = g(x,t)$.

Problem I: *Let t be a fixed time parameter. Given $g(x,t) = \mathbf{U}_t f(x)$, find $f(x)$.*

The inversion problem is similar to the problem of Deconvolution, and the RMP algorithm to be outlined is similar to the algorithm presented in Sect. 10.1.2. Assume that the sought-for initial temperature distribution function $f(x)$ is $N-$periodic and belongs to C^r. Let $t > 0$ be a fixed time parameter. The data function $g(x,t) = \mathbf{U}_t f(x)$ is sampled on the uniform grid $\{k\},\ k = 0, \ldots, N-1,\ N = 2^j$ and the samples are available up to some errors. Thus, we have at our disposal a known time moment t and the data vector $\mathbf{z} = \{z[k] = g(k,t) + \varepsilon_k\},\ k = 0, \ldots, N-1$. The function $f(x)$ is approximated by a spline $S(x) \in {}^{2r}\mathscr{S}$, which is a combination of the spline wavelet packets.

10.3.1 Dictionaries for Heat Equation RMP

As in Sect. 10.2.1, different dictionaries $\mathbf{D}_A$ and $\mathbf{D}_T$ are used for approximation of the solution and for testing the reminders, respectively.

The dictionary $\mathbf{D}_A$ is formed from one-sample translations of the spline wavelet packets $\psi_{m,l}^{2r}(t) \in {}^{2r}\mathscr{W}_{m,l}$,

$$\mathbf{D}_A = \left\{\psi^{2r}_{m,l,k}(x)\right\}_{k=0}^{N-1}, \tag{10.16}$$

$$\psi^{2r}_{m,l,k}(t) = \psi^{2r}_{m,l}(x+k) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n]\, \omega^{kn}\, \gamma^{2r}[n](x), \quad k = 0, \ldots, N-1.$$

The available data (up to noise) is the sampled function $g(x,t)$, which is the temperature distribution at the moment t, provided the initial distribution was $f(x)$. Therefore, once the function $f(x)$ is approximated by spline wavelet packets $\psi^{2r}_{m,l}(x+k)$, it is natural to test the available data by sampled waveforms $\theta_{m,l}(x+k,t)$, which approximately present the temperature distribution at the moment t, provided the initial distribution was $\psi^{2r}_{m,l}(x+k)$. To design the testing dictionary, we utilize schemes, which are based either on the difference or the collocation approximation of the heat equation.

Difference approximation: Denote by $\theta^d_{m,l}(x,t)$ the spline from the space ${}^{2r}\mathscr{S}$ (with respect to x), which is a solution to the difference approximation of the heat equation

$$\frac{\partial \theta^d_{m,l}(x,t)}{\partial t} = \delta^2_x[\theta^d_{m,l}](x,t), \quad \theta^d_{m,l}(x,0) = \psi^{2r}_{m,l}(x).$$

Here, $\delta^2_x[q](x) \stackrel{\text{def}}{=} (q(x-1) - 2q(x) + q(x+1)$ is the circular second central difference.

Collocation approximation: Denote by $\theta^c_{m,l}(x,t)$ the spline from the space ${}^{2r}\mathscr{S}$ (with respect to x), which satisfies the collocation conditions for the heat equation

$$\frac{\partial \theta^c_{m,l}(k,t)}{\partial t} = \left(\theta^c_{m,l}\right)''_x(k,t),\ k = 0, \ldots, N-1,\ \ \theta^c_{m,l}(x,0) = \psi^{2r}_{m,l}(x).$$

Equation (7.25) provides the following representation for the splines $\theta^d_{m,l}(x,t)$ and $\theta^c_{m,l}(x,t)$:

$$\theta^{d/c}_{m,l}(x,t) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n](t)\, \gamma^{2r}[n](x), \text{ where}$$

$$v_{m,l}[n](t) = v_{m,l}[n]\, \mu(t)[n], \quad \mu(t)[n] \stackrel{\text{def}}{=} \begin{cases} e^{-a[n]\,t}, & \text{for } \theta^d_{m,l} \\ e^{-b^r[n]\,t}, & \text{for } \theta^c_{m,l}, \end{cases} \tag{10.17}$$

$$a[n] \stackrel{\text{def}}{=} \left(2\sin\frac{\pi n}{N}\right)^2, \quad b^r[n] \stackrel{\text{def}}{=} \left(2\sin\frac{\pi n}{N}\right)^2 \frac{u^{2(r-1)}[n]}{u^{2r}[n]}.$$

Denote by $\Theta_{m,l} = \left\{\theta^{d/c}_{m,l}(\kappa,t)\right\}$, $k = 0, \ldots N-1$, the sampled spline $\theta^{d/c}_{m,l}(x,t)$. Equation (4.30) provides an explicit representation

$$\Theta_{m,l}[\kappa] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{\kappa n} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} \mu(t)[n]\, v_{m,l}[n] \quad \kappa = 0, \ldots N-1.$$

The testing dictionary $\mathbf{D}_T$ is constituted by one-sample circular shifts of the signals $\Theta_{m,l}$: $\mathbf{D}_T = \left\{\Theta_{m,l,k}(\kappa) = \Theta_{m,l}[\kappa + k)\right\},\ \kappa = 0, \ldots, N-1.$

$$\Theta_{m,l,k}(\kappa) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{(\kappa+k)n} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} \mu(t)[n]\, v_{m,l}[n], \tag{10.18}$$

where $\kappa, k = 0, \ldots, N-1$, and the sequence $\mu(t)[n]$ is defined in Eq. (10.17).

Transformation of the wavelet packet $\psi^{2r}_{m,l}(x) \in {}^{2r}\mathscr{S}$ into the spline $\theta^{d/c}_{m,l}](x,t)$ from the same space ${}^{2r}\mathscr{S}$ is similar to convolving it with another spline, whose SHA spectrum is condensing around zero when t is increasing. Depending on t, overlap of SHA spectra $\left\{v_{\tilde{m},\tilde{l}}[n]\right\}$ of some high-frequency wavelet packets $\psi^{2r}_{\tilde{m},\tilde{l}}(x)$ with the set $\{\mu(t)[n]\}$ becomes negligible. Thus, the norms

$$\left\|\Theta_{\tilde{m},\tilde{l}}\right\| = \frac{1}{N} \sqrt{\sum_{n=-N/2}^{N/2-1} \left| \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} \mu(t)[n]\, v_{\tilde{m},\tilde{l}}[n] \right|^2} \approx 0.$$

Such wavelet packets $\psi^{2r}_{\tilde{m},\tilde{l}}(x)$ are discarded from the dictionary $\mathbf{D}_A$, and the corresponding testing signals $\Theta_{\tilde{m},\tilde{l},\kappa}$, $\kappa = 0, \ldots, N-1$, are discarded from the dictionary $\mathbf{D}_T$.

The RMP algorithm for inversion of the heat equation is implemented by the same steps as the RMP for Deconvolution.

10.3.2 Computational Scheme

10.3.2.1 Preprocessing

Dictionaries: The depth M and the order $2r \geq s+1$ of the approximating spline wavelet packets are chosen basing on a prior information on the smoothness of the solution $f(x) \in C^s$. Their SHA spectra are calculated using Eq. (8.49):

$$\psi^{2r}_{m,l}(x) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} v_{m,l}[n]\, \gamma^{2r}[n](x) \in {}^{2r}\mathscr{S} \subset C^s,$$
$$r = 0, \ldots, M,\ l = 0, \ldots, 2^m - 1.$$

Then, the approximation dictionary $\mathbf{D}_A$ is presented by Eq.(10.16).

Elements $\Theta_{m,l,k}$ of the testing dictionary $\mathbf{D}_T$, which is defined by Eq.(10.18), are derived rom the coordinates $\mu(t)[n]$ given by Eq. (10.17) and $v_{m,l}[n]$.

Modeling the noise: We assume that the error vector $\mathbf{e}$ is a zero mean Gaussian white noise. The Fourier coefficients of the function $g(x,t)$ are given in Eq. (7.17):

$$g_n(t) = c_n(f)\, e^{-t(2\pi\, n/N)^2}. \tag{10.19}$$

Their effective support is shrinking fast as t is growing. Due to Eq. (3.8), the same can be said about the DFT of its sampling $\left\{\hat{g}[n](t)\right\}$, $\hat{g}[n](t) \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \omega^{-kn}\, g(k,t)$. Therefore, the array $\{g(k,t)\}$ is efficiently bandlimited. That means, its significant DFT coefficients $\hat{g}[n](t)$ are located inside the band $-K(t) < n < K(t)$, where $K(t) < N/2$. The width $K(t)$ can be evaluated from Eqs. (10.19) and (3.8). Hence, the DFT coefficients of the data vector $\mathbf{z}$

$$\left\{\hat{z}[n]\right\}_{n\in[K(t),N/2-1]\cup[-N/2,-K(t)]} \approx \left\{\hat{e}[n]\right\}_{n\in[K(t),N/2-1]\cup[-N/2,-K(t)]},$$

and we simulate the DFT of noise by symmetric expansion of $\left\{\hat{z}[n]\right\}$ from the subset $[K(t),\, N/2-1]\cup[-N/2,\, -K(t)]$ to the whole frequency domain $[-N/2+1,\, N/2]$. Thus, noise is modeled by the vector $\tilde{\mathbf{e}} = \{\tilde{e}_i\}_{i=0}^{N-1}$, whose DFT spectrum $\widehat{\tilde{e}}[n] = \hat{z}[n]$ as $n \in [K(t),\, N/2-1] \cup [-N/2,\, -K(t)]$.

Setting threshold and reduction of dictionaries: Calculate the inner products $\left|\langle \Theta_{m,l,k},\, \mathbf{z}\rangle\right|$ and $\left|\langle \Theta_{m,l,k},\, \tilde{\mathbf{e}}\rangle\right|$, which are needed for the oblique projections (10.11), as

$$T_{m,l,k}(\mathbf{z}) \stackrel{\text{def}}{=} \left|\langle \Theta_{m,l,k},\, \mathbf{z}\rangle\right| = \frac{1}{N}\left|\sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\, \mu(t)[n]\, v_{m,l}[n]\, \hat{z}[n]^*\right|,$$

$$e_{m,l,k} \stackrel{\text{def}}{=} \left|\langle \Theta_{m,l,k},\, \tilde{\mathbf{e}}\rangle\right| = \frac{1}{N}\left|\sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\, \mu(t)[n]\, v_{m,l}[n]\, \widehat{\tilde{e}}[n]^*\right|.$$

Calculate the oblique projection coefficients

$$\acute{\alpha}_{m,l,k}(\mathbf{z}) = \frac{T_{m,l,k}(\mathbf{z})}{\|\Theta_{m,l}\|}\left(1 - \frac{e_{m,l,k}}{\left|T_{m,l,k}(\mathbf{z})\right|}\right)_+$$

and the values $A_{m,l}(\mathbf{z}) \stackrel{\text{def}}{=} \max|\acute{\alpha}_{m,l,k}(\mathbf{z})|$ with respect to $k = 0, \ldots, N-1$. .

Set the threshold T_s and reduce dictionaries $\mathbf{D}_T$ and $\mathbf{D}_A$ as it is described in Sect. 10.2.4.

Initialization of the algorithm: The initial temperature distribution $f(x)$ from Eq. (10.15) is approximated by a sequence of splines

$$f^j(x) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma^j[n]\, \gamma^{2r}[n](x) \in {}^{2r}\mathscr{S},$$

which are combinations of the wavelet packets from the reduced dictionary $\tilde{\mathbf{D}}_A$.

Define the initial approximation $f^0(x) \equiv 0$ and the initial remainder $\mathbf{z}^0(t) \stackrel{\text{def}}{=} \mathbf{z}$. Then, we have

$$T_{m,l,k}(\mathbf{z}^0) = T_{m,l,k}(\mathbf{z}), \quad \acute{\alpha}_{m,l,k}(\mathbf{z}^0) = \acute{\alpha}_{m,l,k}(\mathbf{z}), \quad \sigma^0[n] = 0.$$

Put $j = 0$.

Pursuit:

1. Find $\tilde{m}, \tilde{l}, \tilde{k} = \arg\max_{m,l,k} \left|\grave{\alpha}_{m,l,k}(\mathbf{z}^j)\right|$.
2. Denote $\lambda^{j+1} \stackrel{\text{def}}{=} \acute{\alpha}_{\tilde{r},\tilde{l},\tilde{k}} / \left\|\Theta_{\tilde{r},\tilde{l}}\right\|$.
3. Put $\sigma[n]^{j+1} = \sigma[n]^j + \lambda^{j+1}\, v_{\tilde{r},\tilde{l}}[n]\, \omega^{\tilde{k}n},\ n = 0, \ldots, N-1$.
4. Put $\hat{z}^{j+1}[n] = \hat{z}^j[n] - \lambda^{j+1}\, Q^{2r,2q}[n]\, \mu[n]\, v_{\tilde{r},\tilde{l}}[n]\, \omega^{\tilde{k}n},\ n = 0, \ldots, N-1$.
5. Calculate the inner products

$$T_{m,l,k}(\mathbf{z}^{j+1}) = \frac{1}{N} \left| \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}\, \mu(t)[n]\, v_{m,l}[n]\, \hat{z^{j+1}}[n]^* \right|.$$

6. Calculate the oblique projection coefficients

$$\grave{\alpha}_{m,l,k}(\mathbf{z}^{j+1}) = \frac{T_{m,l,k}(\mathbf{z}^{j+1})}{\left\|\Theta_{m,l}\right\|} \left(1 - \frac{e_{m,l,k}}{\left|T_{m,l,k}(\mathbf{z}^{j+1})\right|}\right)_+$$

   and $A_{m,l}(\mathbf{z}^{j+1}) \stackrel{\text{def}}{=} \max |\acute{\alpha}_{m,l,k}(\mathbf{z}^{j+1})|$ with respect to $k = 0, \ldots, N-1$.
7. If for some $(\bar{m}, \bar{l})$ the value $A_{\bar{m},\bar{l}}(\mathbf{z}^{j+1}) < T_s$ then remove $\psi^{2r}_{\bar{m},\bar{l},k}(t)$ and $\Theta_{\bar{m},\bar{l},k},\ k = 0, \ldots N-1$, from the dictionaries $\mathbf{D}_A$ and $\mathbf{D}_T$, respectively.

If the dictionaries are not empty then $j = j + 1$, go to 1.
Otherwise: the approximate solution

$$f(x) \approx f^j(x) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma^j[n]\, \gamma^{2r}[n](x).$$

Calculate the grid points samples of the approximate solution by IFFT using Eq. (4.30):

$$f^j(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}} \sigma^j[n]. \tag{10.20}$$

If need be, we calculate the values of the spline $f^j(t)$ at dyadic or triadic rational points by the subdivision schemes described in Sects. 6.2 and 6.3.

Like the Deconvolution, the above algorithm is implemented by the MATLAB code `RMP_runP.m` in an interactive way for either the difference or the collocation approximation.

10.4 Examples

We display results of a few experiments to test the RMP Deconvolution algorithm that was applied to blurred natural signals affected by a Gaussian white noise. As a source for the test signals the benchmark images such as "Barbara", "Lena" and "Fingerprint" were used. Each column was processed separately as a 1D signal. The stopping threshold T_s and the reduced dictionaries were defined based on properties of a single randomly selected column. Then, the number of iterations for each signal (column) was determined automatically. Separate processing of 512 columns of length 512 required from a dozen of seconds to 3 minutes in the Matlab environment, depending on the strength of the blurring. Strong blurring means that the effective bandwidth of the kernel's spectrum (the passband of the filter) is narrow. Thus, the reduced dictionaries $\tilde{\mathbf{D}}_A$ and $\tilde{\mathbf{D}}_T$ contain relatively small number of waveforms. In this case, processing is faster compared to the case when blurring is weak.

Another series of experiments was conducted on inversion of the heat equation in presence of a Gaussian white noise.

The quality of images restoration is evaluated visually and by the peak signal-to-noise ratio (PSNR) in decibels

$$PSNR \stackrel{\text{def}}{=} 10 \log_{10} \left(\frac{N\, 255^2}{\sum_{k=1}^{N} (x_k - \tilde{x}_k)^2} \right) dB.$$

Here, $\{x_k\}_{k=1}^N$ are the samples of the original image and $M = \max_k |x_k|$, while $\{\tilde{x}_k\}_{k=1}^N$ are the samples of the restored image. In addition, we display restoration of separate columns.

All the subsequent images, which illustrate the results of the experiments were produced by the MATLAB code `RMP_runP.m`.

Fig. 10.5 The original "Barbara" image

Example: Deconvolution of the "Barbara" image affected by moderate blurring and strong noise:
In this experiment, columns of the "Barbara" image (Fig. 10.5) were blurred using Gaussian kernel whose STD $\sigma_k = 2$ and corrupted by Gaussian white noise whose STD was $\sigma_n = 20$.

Figure 10.1 displays the blurring kernel, its Fourier spectrum and the column # 450 from the blurred and noised "Barbara" image versus the original column. That figure, together with Fig. 10.2 illustrate the process of noise modeling for that column.

For the Deconvolution, we use the spline wavelet packets of fourth order (cubic splines) from levels 1 to 6. Altogether there are 126 wavelet packets $\psi^4_{m,l}$, $m = 1, \ldots, 6$, $l = 0, \ldots, 2^m - 1$. Respectively, there are 126 testing signals $\Theta_{m,l} = \left\{ \chi \circledast \psi^4_{m,l}(k) \right\}$, $k = 0, \ldots, 511$. Setting the stopping threshold and the process of reduction of dictionaries is described in Sect. 10.2.4 and is illustrated in Fig. 10.4. In this example, the stopping threshold $T_s = 1$ and the dictionaries consist of shifts of 29 waveforms.

Results of the application of the RMP algorithm to columns of the corrupted image are displayed in Figs. 10.6, 10.7 and 10.8. The left frame in Fig. 10.6 displays the blurred and noised image. Its PSNR with respect to the original image is 20.86 dB. The right frame displays the restored image. Its PSNR with respect to the original image is 23.73 dB. We observe that noise is significantly reduced and the fine texture of the image is partially reconstructed.

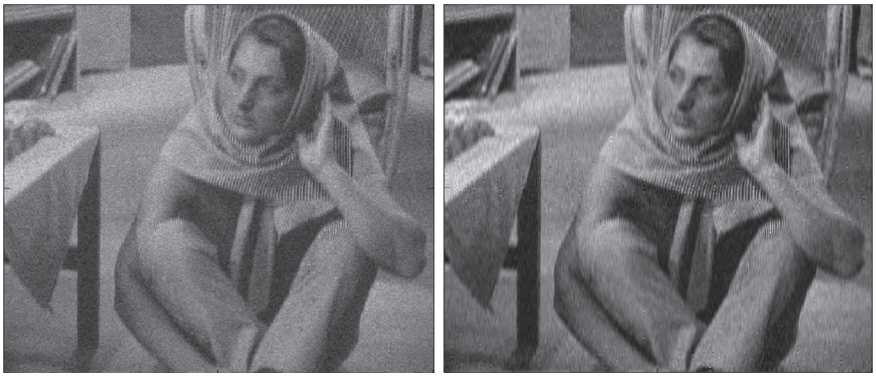


Fig. 10.6 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 2$) and noised (STD = 20) input

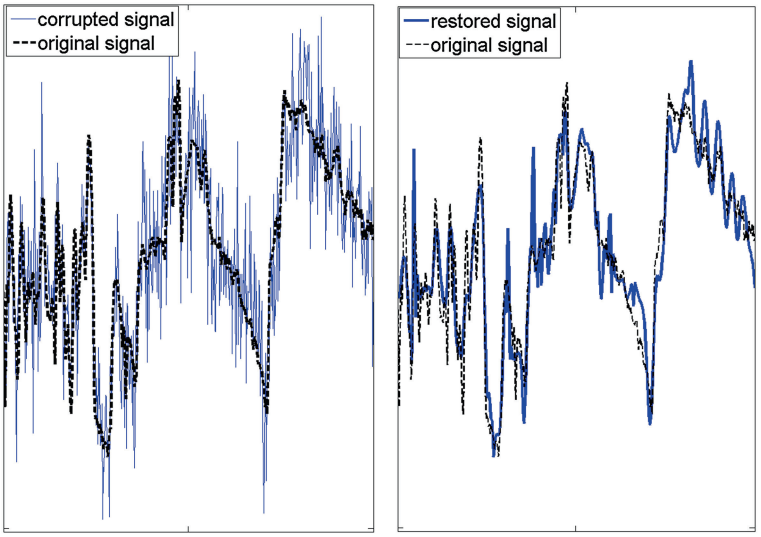


Fig. 10.7 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 2$) and noised (STD = 20) input. Column #450

Figures 10.7 and 10.8 illustrate reconstruction of single columns #450 and #30, respectively. The left frames in the figures display the blurred and noised column (solid line) versus the original column (dashed line). The right frames display the restored original column (solid line) versus the original column (dashed line).

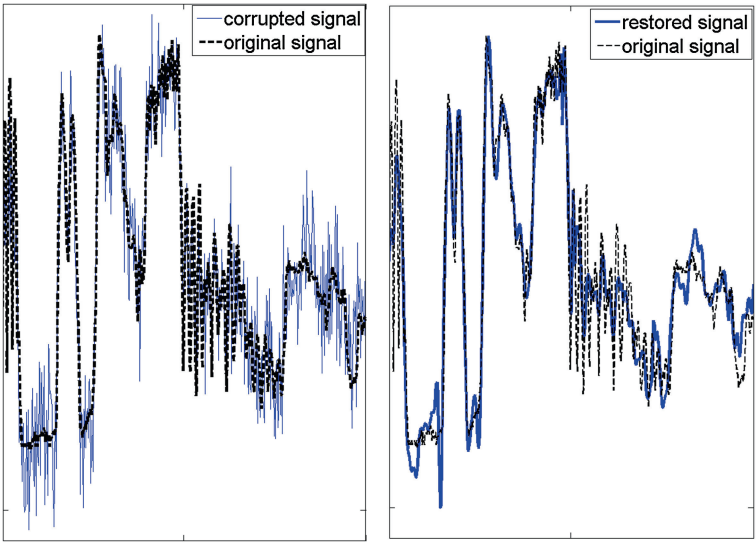


Fig. 10.8 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 2$) and noised (STD = 20) input. Column #30

Example: Deconvolution of the "Barbara" image affected by stronger blurring, no noise added:
In this experiment, columns of the "Barbara" image (Fig. 10.5) were blurred using the Gaussian kernel whose STD $\sigma_k = 3$ and no noise was added. The image was restored using the cubic spline wavelet packets from the first to the sixth decomposition levels. The stopping threshold T_s was set to 0.1. Altogether, 37 waveforms were involved.

Results of the application of the RMP algorithm to columns of the corrupted image are displayed in Figs. 10.9, 10.10 and 10.11. The left frame in Fig. 10.9 displays the blurred image. Its PSNR with respect to the original image is 25.17 dB. The right frame displays the restored image. Its PSNR with respect to the original image is 29.86 dB. One can observe that the fine texture of the image is reconstructed almost completely. Figures 10.10 and 10.11 illustrate reconstruction of single columns #450 and #70, respectively. The left frames in the figures display the blurred columns (solid lines) versus the original columns (dashed lines). The right frames display the restored columns (solid lines) versus the original columns (dashed lines). We observe that, despite a strong blurring, the columns are restored almost perfectly.

Example: Inversion of heat equation whose initial condition was the "Lena" image. Moderate evolution time and weak noise:
In this experiment, columns of the "Lena" image (Fig. 10.12), which serve as initial conditions for 1D heat equation, were blurred in process of time evolution $t = 10$ and affected by a weak Gaussian white noise whose STD was $\sigma_n = 2$. The input PSNR was 27.56 dB. As in previous examples, we utilized the spline wavelet packets

Fig. 10.9 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 3$) input

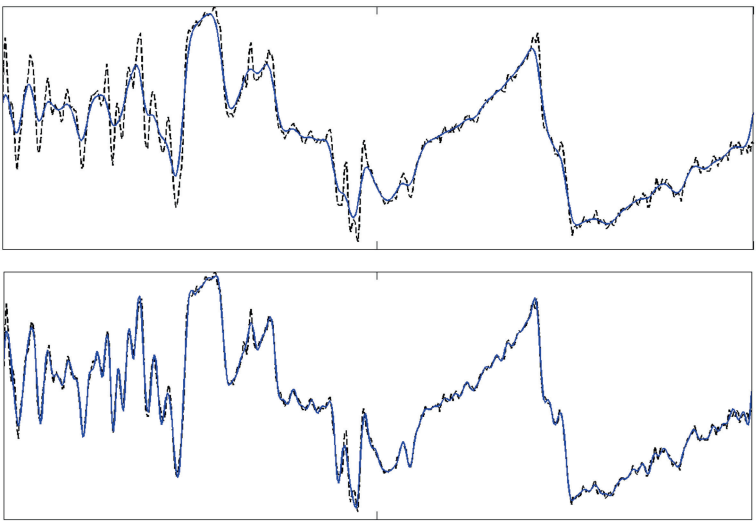


Fig. 10.10 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 3$) input. Column #450. *Top* the blurred column. *Bottom* the column restored

of fourth order (cubic splines) from levels 1 to 6: $\psi^4_{m,l},\ m = 1, \ldots, 6,\ l = 0, \ldots, 2^m - 1$. The width of the band for noise modeling was chosen to be 450, the stopping threshold $T_s = 0.05$, and the dictionaries consisted of shifts of 23 waveforms. In this example, the collocation approximation of the heat equation was used.

Results of the application of the RMP algorithm to columns of the corrupted image are displayed in Figs. 10.13, 10.14 and 10.15. The left frame in Fig. 10.13 displays the blurred and noised image whose PSNR is 27.56. The right frame displays the

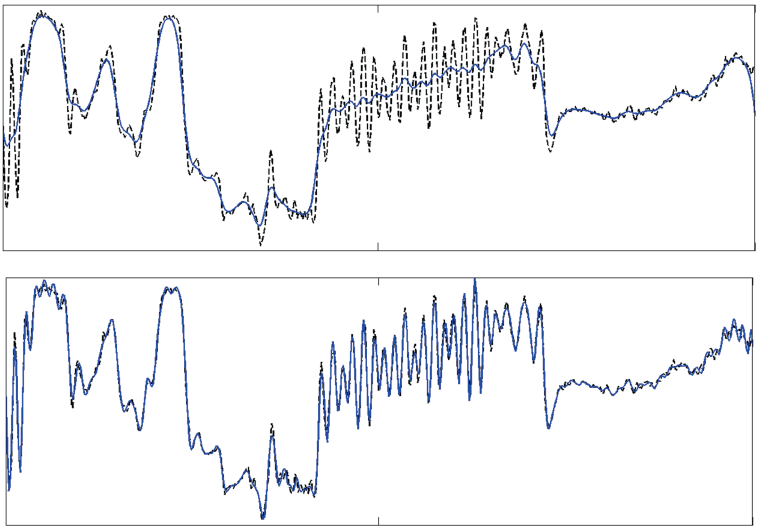


Fig. 10.11 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 3$) input. Column #70. *Top* the blurred column. *Bottom* the column restored

Fig. 10.12 The original "Lena" image

Fig. 10.13 Inversion of the heat equation whose initial conditions were columns of the "Lena" image. Evolution time $t = 10$. Input is affected by Gaussian noise (STD = 2)

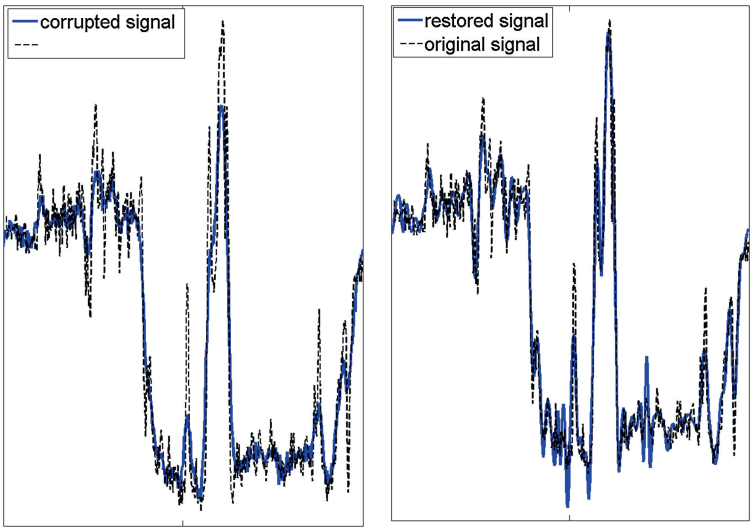


Fig. 10.14 Inversion of the heat equation whose initial condition was column #200 of the "Lena" image. Evolution time $t = 10$. Input is affected by Gaussian noise (STD = 2)

restored image. Its PSNR with respect to the original image is 29.24 dB. We observe that noise is completely removed and, visually, the restored image is sharper than the blurred one.

Figures 10.14 and 10.15 illustrate reconstruction of single columns #200 and #300, respectively. The left frames display the blurred and noised column (solid line) versus the original column (dashed line). The right frames display the restored column (solid line) versus the original column (dashed line). We observe that the structure of the columns is successfully extracted from blurred and noised input.

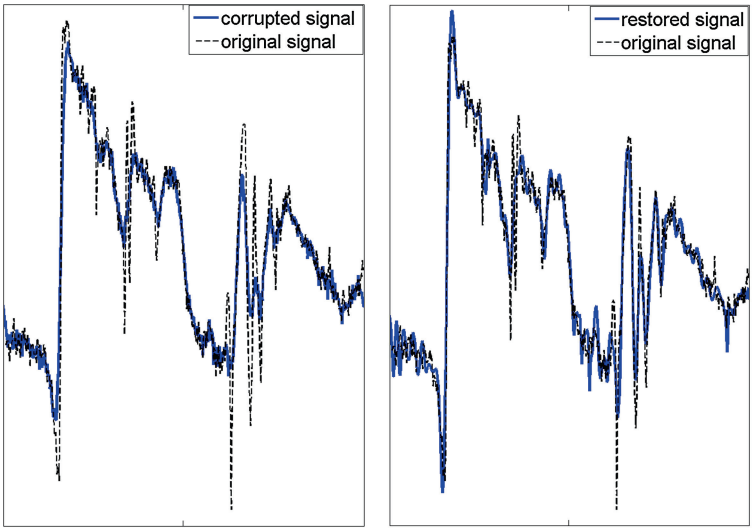
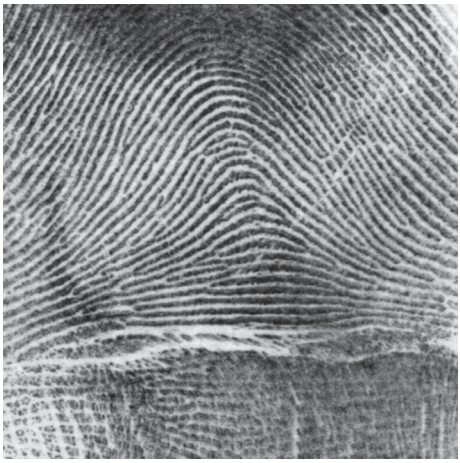


Fig. 10.15 Inversion of the heat equation whose initial condition was column #300 of the "Lena" image. Evolution time $t = 10$. Input is affected by Gaussian noise (STD = 2)

Fig. 10.16 The original "Fingerprint" image



Example: Restoration of the "Fingerprint" image from strongly noised input: In this experiment, columns of the "Fingerprint" image (Fig. 10.16) are buried in strong Gaussian white noise whose STD is $\sigma_n = 50$. No blurring is applied. The input PSNR is 14.14. We utilized the spline wavelet packets of fourth order (cubic splines) from levels 1 to 6: $\psi^4_{m,l}$, $m = 1, \ldots, 6$, $l = 0, \ldots, 2^m - 1$. The width of the band for noise modeling was chosen to be 400, the stopping threshold being $T_s = 1$, which reduced the dictionaries to 52 waveforms.

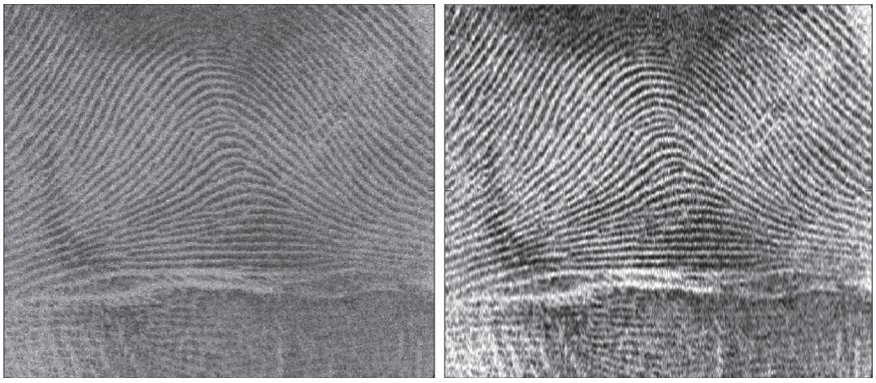


Fig. 10.17 Restoration of the "Fingerprint" image from the input affected by Gaussian noise (STD = 50)

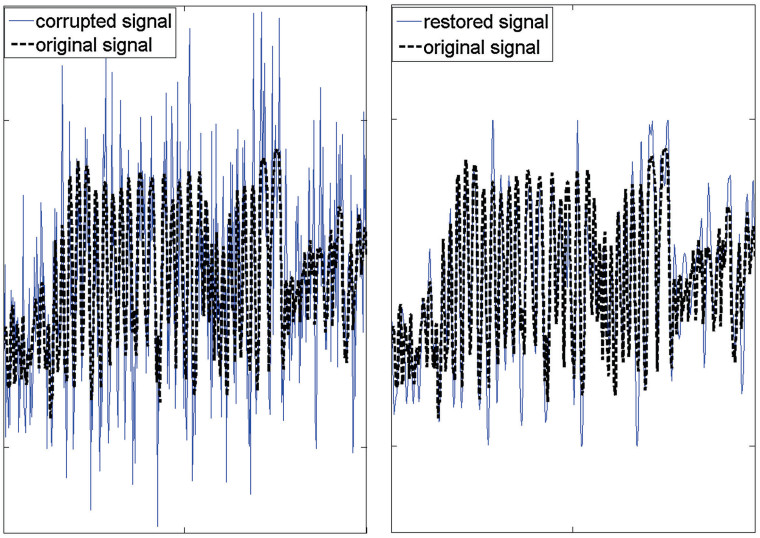


Fig. 10.18 Restoration of column #250 of the "Fingerprint" image from the input affected by Gaussian noise (STD = 50)

Results of the application of the RMP algorithm to columns of the corrupted image are displayed in Figs. 10.17, 10.18 and 10.19. The left frame in Fig. 10.17 displays the noised image whose PSNR is 14.14 dB. Right frame displays the restored image. Its PSNR with respect to the original image is 19.61 dB. We observe that noise is significantly reduced and the texture of the image is revealed.

Figures 10.18 and 10.19 illustrates reconstruction of the single columns #250 and #300, respectively. The left frames display the noised column (solid line) versus the original column (dashed line). Right frames display the restored column (solid

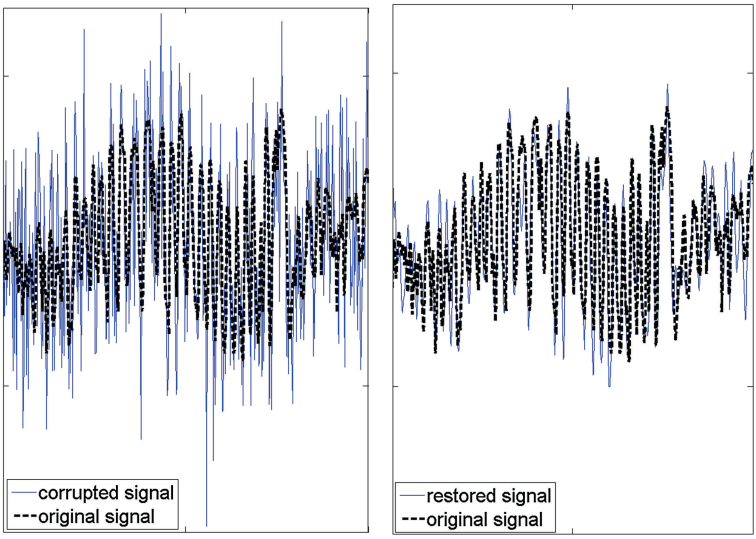


Fig. 10.19 Restoration of column #300 of the "Fingerprint" image from the input affected by Gaussian noise (STD = 50)

line) versus the original column (dashed line). We observe that the structure of the columns is successfully extracted from the noised input.

10.4.1 Conclusions

The regularized matching pursuit is an efficient algorithm to restore signals from noised blurred discrete data. The computational engine, which enables an efficient implementation of the algorithm, was provided by the Spline Harmonic Analysis (SHA). The operations of convolution and finding the projection coefficients become straightforward. The usage of splines enables us to map the discrete noised data into the spaces of continuous functions. Experiments confirmed the efficiency of the algorithm. The algorithm was able to extract the coherent structure of signals, which were subjected to strong blurring and affected by noise. It provides sparse representations of solutions.

However, direct extension of the algorithm to image processing appears computationally expensive. A somewhat related fast algorithm for image processing, which is based on 2D spline wavelet packets, was developed in [6]. It is presented in Chapter 11.

References

1. A.N. Tikhonov, Solution of incorrectly formulated problems and the regularization method. Soviet Math. Dokl. **4**(4), 1035–1038 (1963)
2. A.N. Tikhonov, V.Y. Arsenin, *Solutions of ill-posed problems* (John Wiley & Sons, New York, 1977)
3. S. Mallat, Z. Zhang, Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. **41**(12), 3397–3415 (1993)
4. G. Davis, S. Mallat, M. Avellaneda, Adaptive greedy approximations. Constr. Approx. **13**(1), 57–98 (1997)
5. Y.C. Pati, R. Rezaiifar, P.S. Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems and Computers. vol. 1 (1993), pp. 40–44
6. A. Averbuch, V. Zheludev, P. Neittaanmäki, J. Koren, Block based deconvolution algorithm using spline wavelet packets. J. Math. Imaging Vision **38**(3), 197–225 (2010)

Chapter 11
Block-Based Inversion of the Heat Equations

Abstract This chapter presents robust methods, which refine the algorithms, in Sect. 7.2, for inversion of the heat equations. The idea behind the algorithms is to solve the inversion problem separately in different frequency bands. This is achieved by using spline wavelet packets. The solutions that minimize some parameterized quadratic functionals, are derived as linear combinations of the wavelet packets. Choice of parameters, which is performed automatically, determines the trade-off between the solution regularity and the initial data approximation. The Spline Harmonic Analysis (SHA) technique provides a unified computational scheme for the fast implementation of the algorithm and an explicit representation of the solutions. The presented algorithms provide stable solutions that accurately approximate the initial temperature distribution.

In this chapter we describe one more application of the spline wavelet packets. In Chap. 7, algorithms for solving two related ill-posed problems were presented. These problems, *deconvolution* and *inversion of the heat equations*, were dealt with by using a spline version of the Tikhonov regularization. According to the Tikhonov regularization methodology, an approximate solution is derived as a minimizer of a parameterized functional consisting of two components: a discrepancy functional, which measures the approximation of the known data, and a stabilizer, which controls the regularity of the solution. The numerical regularization parameter provides a trade-off between the approximation and the solution's regularity. This parameter is derived automatically. Its value depends on the relative shares of the coherent signal and noise in the available data.

These shares are different for different frequency components of the data. To take this into account, it is natural to solve the problems separately in different frequency bands, while the regularization parameters according to the signal-to-noise ratio (SNR) are to be found at each band. By this means, the adaptation abilities and the robustness of the method can be extended.

In this chapter we deal with the inversion problem. The approach to the solution of the deconvolution problem is quite similar (see [2]).

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_11,

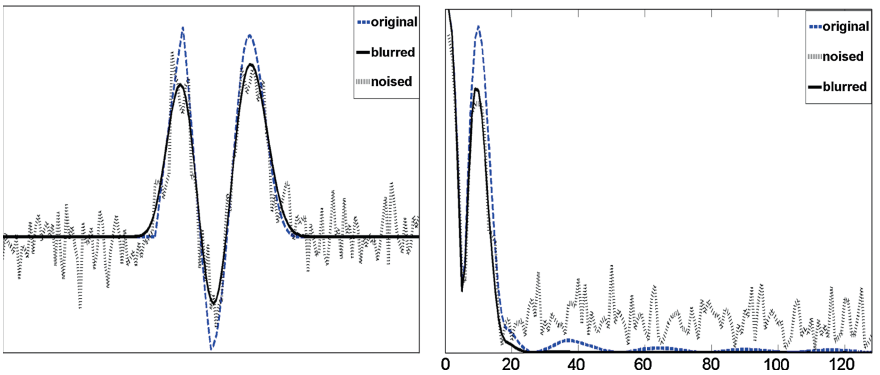


Fig. 11.1 *Left* original temperature distribution $f(x)$—*dashed line*; temperature distribution $g(x,t)$ after some time t—*solid line*; available noised data $\mathbf{z}$—*dotted line*. *Right* respective Fourier spectra

Figure 11.1 illustrates SNR in different frequency domains. We observe that, while the influence of noise on low-frequency component of the signal is insignificant, the high-frequency domain of the data is completely occupied by noise.

The orthonormal spline wavelet packets coupled with the SHA technique provide a practical tool for implementation of this approach. These wavelet packets, which are described in Chap. 8, produce a split of the frequency domain of a signal (image) into a set of bands whose overlap is minimal. The Best Basis methodology [4, 6] outlined in Chap. 8 enables us to find an optimal partition of the frequency domain for a given signal (image). Another method to select essential frequency bands is the so-called Block Pursuit algorithm (BPA). This method, which, to some extent, is related to the Matching Pursuit ([5] and Chap. 10), enables us, in several cases, to achieve a superior restoration quality at the expense of higher computational cost compared to the Best Basis scheme. The solutions are explicitly represented by splines. Their values at integer grid points are calculated by application of the inverse fast Fourier transform (FFT), while the values at diadic or triadic rational points can be calculated by using the subdivision algorithms presented in Chap. 6.

11.1 One-Dimensional Block-Based Heat Equation Inversion

11.1.1 Preliminaries

In this section we discuss approximate inversion of the one-dimensional heat equation

$$\frac{\partial g(x,t)}{\partial t} = g''_x(x,t),\ g(x,0) = f(x) \Longleftrightarrow g(x,t) = \mathbf{U}_t f(x). \tag{11.1}$$

Problem IP1 Let t be a fixed time parameter. Given $g(x,t) = \mathbf{U}_t f(x)$, find $f(x)$.

If $f(x)$ is a T-periodic function then there exists explicit expression (Eq. 7.18) of the solution via the Fourier series. However, this theoretical solution is unstable due to fast decay of the Fourier coefficients. In practice, only sampled data, which, typically, is corrupted by noise, is available.

In Sect. 7.2.1 a method for stable approximate solution of the inversion problem was described. The method consists of embedding the problem into a spline setting. This is followed by running the Tikhonov regularization scheme in the spline space. In this section this method is refined using the wavelet packets machinery.

As in Sect. 7.2.1, we assume that the sought-for initial temperature distribution function $f(x)$ is $N-$periodic and belongs to C^r. Let $t > 0$ be a fixed time parameter. The data function $g(x,t) = \mathbf{U}_t f(x)$ is sampled on the uniform grid $\{k\}$, $k = 0, \ldots, N-1$, $N = 2^j$, and the samples are available up to some errors. Thus, we have at our disposal a known time moment t and the data vector $\mathbf{z} = \{z[k] = g(k,t) + \varepsilon_k\}$, $k = 0, \ldots, N-1$. The function $f(x)$ is approximated by a spline $S(x) \in {}^{2r}\mathscr{S}$, which is a linear combination of spline wavelet packets.

Embedding the problem into a spline setting is implemented by spline approximation of the heat equation, which is achieved in two ways.

Difference approximation: Denote, by $\mathbf{V}_t^d$, the linear operator defined on the spline space ${}^{2r}\mathscr{S}$ such that for $S(x) \in {}^{2r}\mathscr{S}$, $\mathbf{V}_t^d S(x) = S(x,t)$, where $S(x,t)$ is the spline from ${}^{2r}\mathscr{S}$ (with respect to x), which provides a solution to the difference approximation of the heat equation

$$\frac{\partial S(x,t)}{\partial t} = \delta_x^2[s](x,t), \quad S(x,0) = S(x).$$

Here, $\delta_x^2[q](x) \stackrel{\text{def}}{=} q(x-1) - 2q(x) + q(x+1)$ is the circular second central difference.

Collocation approximation: Denote, by $\mathbf{V}_t^c$, the linear operator on ${}^{2r}\mathscr{S}$ such that for $S(x) \in {}^{2r}\mathscr{S}$, $\mathbf{V}_t^c S(x) = S(x,t)$, where $S(x,t)$ is the spline from ${}^{2r}\mathscr{S}$, which satisfies the collocation conditions for the heat equation

$$\frac{\partial S(k,t)}{\partial t} = s_x''(k,t), \; k = 0, \ldots, N-1, \quad S(x,0) = S(x).$$

If the spline $S(x)$ is expanded over the exponential orthonormal basis

$$S(x) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma[n]\, \gamma^{2r}[n](x),$$

then the splines $S(x,t) = \mathbf{V}_t^{d(c)} S(x)$ are

$$S(x,t)=\frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\sigma(t)[n]\,\gamma^{2r}[n](x),\quad \sigma(0)[n]=\sigma[n], \tag{11.2}$$

$$\sigma[n](t)=\sigma[n]\,\eta[n](t),\quad \eta[n](t)\stackrel{\text{def}}{=}\begin{cases} e^{-a[n]\,t}, & \text{when } \mathbf{V}_t=\mathbf{V}_t^d \\ e^{-b^r[n]\,t}, & \text{when } \mathbf{V}_t=\mathbf{V}_t^c, \end{cases}$$

$$a[n]\stackrel{\text{def}}{=}\left(2\sin\frac{\pi n}{N}\right)^2,\quad b^r[n]\stackrel{\text{def}}{=}\left(2\sin\frac{\pi n}{N}\right)^2\frac{u^{2(r-1)}[n]}{u^{2r}[n]}. \tag{11.3}$$

11.1.2 Partial Solution of the Inversion Problem in the Subspace ${}^{2r}\mathscr{W}_{m,l}$

11.1.2.1 Splines from the Subspaces ${}^{2r}\mathscr{W}_{m,l}$

Assume a spline $S(x)\in{}^{2r}\mathscr{S}$ is expanded over the orthonormal bases

$$S(x)=\frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\sigma[n]\,\gamma^{2r}[n](x)=\sum_{k=0}^{N-1}q[k]\varphi^{2r}(x-k),$$

$$\varphi^p(t-k)=\frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\omega^{-nk}\,\gamma^{2r}[n](t),\quad \gamma^{2r}[n](t)=\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\omega^{nk}\,\varphi^p(t-k),$$

$$q[k]=\frac{1}{N}\sum_{n=-N/2}^{N/2-1}\omega^{nk}\,\sigma[n],\quad \sigma[n]=\sum_{k=0}^{N-1}\omega^{-nk}\,q[k]=\hat{q}[n].$$

The orthogonal projection of the spline $S(x)$ onto the subspace ${}^{2r}\mathscr{W}_{m,l}$

$$S_{m,l}(x)=\sum_{k=0}^{N_m-1}q_{m,l}[k]\,\psi_{m,l}^{2r}\left(x-2^m k\right), \tag{11.4}$$

where $\left\{\psi_{m,l}^{2r}\left(x-2^m k\right)\right\}\in{}^{2r}\mathscr{W}_{m,l}\subset{}^{2r}\mathscr{S}$ are the orthonormal wavelet packets.

The wavelet packet $\psi_{m,l}^{2r}(x)$ can be expanded over the orthonormal bases of ${}^{2r}\mathscr{S}$

$$\psi_{m,l}^{2r}(x)=\frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\nu_{m,l}[n]\gamma^{2r}[n](x)=\sum_{k=0}^{N-1}p_{m,l}[k]\varphi^{2r}(x-k), \tag{11.5}$$

$$\nu_{m,l}[n]=\hat{p}_{m,l}[n],\quad \psi_{m,l}^{2r}\left(x-2^m k\right)=\frac{1}{\sqrt{N}}\sum_{n=-N/2}^{N/2-1}\nu_{m,l}[n]\omega^{-2^m kn}\,\gamma^{2r}[n](x)$$

The shift property of the exponential splines Eq. (4.20) was utilized. The coefficients $\nu_{m,l}[n]$ constitute the SHA spectrum of the wavelet packet $\psi^{2r}_{m,l}$.

Substituting Eq. (11.5) into Eq. (11.4), we get

$$\begin{aligned} S_{m,l}(x) &= \sum_{k=0}^{N_m-1} q_{m,l}[k]\, \psi^{2r}_{m,l}\left(x - 2^m k\right) \\ &= \sqrt{\frac{1}{N}} \sum_{k=0}^{N_m-1} q_{m,l}[k] \sum_{n=-N/2}^{N/2-1} \nu_{m,l}[n] \gamma^{2r}[n]\left(x - 2^m k\right) \\ &= \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \nu_{m,l}[n] \gamma^{2r}[n]\,(x) \sum_{k=0}^{N_m-1} \omega^{-2^m kn}\, q_{m,l}[k] \\ &= \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \zeta_{m,l}[n]\, \gamma^{2r}[n]\,(x)\,, \end{aligned}$$

$$\zeta_{m,l}[n] \stackrel{\text{def}}{=} \nu_{m,l}[n] \sum_{k=0}^{N_m-1} \omega^{-2^m kn}\, q_{m,l}[k] = \nu_{m,l}[n]\, \hat{q}_{m,l}[n]_m. \tag{11.6}$$

The coefficients $\left\{\zeta_{m,l}[n]\right\}. n = -N/2, \ldots, N/2-1$, form the SHA spectrum of the projection spline $S_{m,l}(x)$. Recall that $\hat{x}[n]_m = \sum_{k=0}^{N/2^m-1} x[k]\, \omega^{-2^m nk}$ denotes the DFT of signal $\mathbf{x} \in \Pi[N/2^m]$, which is N_m-periodic, where $N_m = N/2^m$.

On the other hand, the SHA spectrum $\zeta_{m,l}[n]$ of the projection spline $S_{m,l}(x)$ can be expressed via the SHA spectrum $\sigma[n]$ of the spline $S(x)$.

Proposition 11.1 *The following representation of the projection coordinates holds*

$$\begin{aligned} \zeta_{m,l}[n] = \nu_{m,l}[n]\, \hat{q}_{m,l}[n]_m &= \frac{\nu_{m,l}[n]}{2^m} \sum_{\lambda=-2^m/2}^{2^m/2-1} \sigma[n + \lambda\, N_m]\, \nu^*_{m,l}[n + \lambda\, N_m] \\ &\approx \frac{1}{2^m}\, \sigma[n]\, \left|\nu_{m,l}[n]\right|^2. \end{aligned} \tag{11.7}$$

The proof is similar to the proof of Proposition 9.4.

Remark 11.1.1 The higher the spline's order $2r$ is, the closer are the coefficients $\zeta_{m,l}[n]$ to $2^{-m}\, \sigma[n]\, \left|\nu_{m,l}[n]\right|^2$.

Equation (11.2) implies that application of the operators $\mathbf{V}_t = \mathbf{V}_t^{d(c)}$ to the spline $S_{m,l}(x)$ produces the spline

$$S_{m,l}(x,t) = \mathbf{V}_t S_{m,l}(x) = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \eta[n](t) \zeta_{m,l}[n]\, \gamma^{2r}[n](x),$$

where the coefficients $\eta[n](t)$ are defined in Eq. (11.3). From Eq. (4.30), samples of the spline $S(x,t) = \mathbf{V}_t S(x)$, which is presented in Eq. (11.2), are

$$y[k] \stackrel{\text{def}}{=} S(k,t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \eta[n](t)\, \sigma[n]\, V^{2r}[n]\, \omega^{kn},$$

$$\hat{y}[n] = \eta[n](t)\, \sigma[n]\, V^{2r}[n], \quad V^{2r}[n] \stackrel{\text{def}}{=} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}.$$

Samples of $S_{m,l}(x,t) = \mathbf{V}_t S_{m,l}(x)$ are

$$y_{m,l}[k] \stackrel{\text{def}}{=} S_{m,l}(k,t) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \eta[n](t)\, \zeta_{m,l}[n]\, V^{2r}[n]\, \omega^{kn}. \tag{11.8}$$

Using Eq. (11.7), we get the approximate relation

$$\begin{aligned} \hat{y}_{m,l}[n] &= \eta[n](t)\zeta_{m,l}[n]\, V^{2r}[n] \\ &\approx \frac{1}{2^m}\, \eta[n](t)\, \sigma[n]\, V^{2r}[n] \left|\nu_{m,l}[n]\right|^2 = \frac{1}{2^m}\, \hat{y}[n] \left|\nu_{m,l}[n]\right|^2. \end{aligned} \tag{11.9}$$

Remark 11.1.2 Equation (11.9) can be interpreted in a sense that confinement of the operator's $\mathbf{V}_t$ domain from the whole spline space ${}^{2r}\mathscr{S}$ to the subspace ${}^{2r}\mathscr{W}_{m,l}$, effectively results in multiplication of the DFT $\hat{y}[n]$ of the sampled output $y[k] = \mathbf{V}_t S(k)$, $k = 0, \ldots, N-1$, with the factor $2^{-m} \left|\nu_{m,l}[n]\right|^2$.

11.1.2.2 Parameterized Spline Solution in the Subspace ${}^{2r}\mathscr{W}_{m,l}$

The scheme of partial solution of Problem IP1 in the subspace ${}^{2r}\mathscr{W}_{m,l}$ is very similar to the scheme of global solution presented in Sect. 7.2.1. By assumption, t is a known time parameter and the vector $\mathbf{z} = \{z[k] = g(k,t) + e_k\} = \mathbf{g} + \mathbf{e}$ is available, where $g(x,t) = \mathbf{U}_t f(x)$. From this data, a partial approximate inversion of the one-dimensional heat Eq. (11.1) is derived as a spline

$$S_{m,l}(x) = \sum_{k=0}^{N_m-1} q_{m,l}[k]\, \psi_{m,l}^{2r}\left(x - 2^m k\right) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \zeta_{m,l}[n]\, \gamma^{2r}[n]\,(x) \tag{11.10}$$

such that the spline $S_{m,l}(x,t) = \mathbf{V}_t S_{m,l}(x)$ approximates, in some sense, the available discrete data $\mathbf{z}$. To be specific, Remark 11.1.2 suggests that the sampled spline $S_{m,l}(k,t)$ should approximate the "filtered" data

$$\tilde{z}_{m,l}[k] \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \hat{\tilde{z}}_{m,l}[n], \text{ where} \tag{11.11}$$

$$\hat{\tilde{z}}_{m,l}[n] \stackrel{\text{def}}{=} \hat{z}[n]\, \frac{\left|\nu_{m,l}[n]\right|^2}{2^m}, \quad \hat{z}[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, z[k],$$

rather than the entire data $\mathbf{z}$. Similarly to Sect. 7.2.1, we find a spline $S(\rho, x) \in {}^{2r}\mathscr{W}_{m,l}$, which minimizes the functional $J_{m,l}(\rho)(S) \stackrel{\text{def}}{=} \rho I(S) + E_{m,l}(S)$, where

$$I(S) \stackrel{\text{def}}{=} \|(S)^{(r)}\|^2, \quad E_{m,l}(S) \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \left(S(k,t) - \tilde{z}_{m,l}[k]\right)^2, \quad S(x,t) \stackrel{\text{def}}{=} \mathbf{V}_t S(x),$$

$\mathbf{V}_t = \mathbf{V}_t^{d(c)}$ and ρ is a numerical parameter.

Assume that a spline $s(x) \in {}^{2r}\mathscr{W}_{m,l}$ is represented as in Eq. (11.10)

$$s(x) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \zeta_{m,l}[n]\, \gamma^{2r}[n]\,(x)\,.$$

Then, Eq. (4.27) implies that

$$I(s) = \int_0^N (s^{(r)}(x))^2\, dx = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left|\zeta_{m,l}[n]\right|^2\, W^{2r,r}[n],$$

where

$$W^{2r,r}[n] \stackrel{\text{def}}{=} \left(2\, \sin \frac{\pi n}{N}\right)^{2r}\, \frac{u^{2r}[n]}{u^{4r}[n]}.$$

Using the DFT representations Eqs. (11.8 and 11.11), we get

$$E_{m,l}(s) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left|\eta[n](t)\zeta_{m,l}[n]\, V^{2r}[n] - \hat{\tilde{z}}_{m,l}[n]\right|^2.$$

Consequently, the parameterized functional

$$J_{m,l}(\rho)(s) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left(\rho \left|\zeta_{m,l}[n]\right|^2\, W^{2r,r}[n] + \left|\eta[n](t)\zeta_{m,l}[n]\, V^{2r}[n] - \hat{\tilde{z}}_{m,l}[n]\right|^2\right)$$

$$= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |\zeta_{m,l}[n]|^2 \left(\rho\, W^{2r,r}[n] + \left| \eta[n](t)\, V^{2r}[n] \right|^2 \right)$$
$$+ \mathscr{R}\rceil \left(\zeta_{m,l}[n]\, \eta[n](t)\, V^{2r}[n]\, \hat{\hat{z}}^*_{m,l}[n] \right) + \left| \hat{\hat{z}}_{m,l}[n]) \right|^2 .$$

The minimum of $J_{m,l}(\rho)(s)$ is achieved when

$$\zeta_{m,l}(\rho)[n] = \frac{\eta^*[n](t)\, V^{2r}[n]\, \hat{\hat{z}}_{m,l}(n)}{A[n](\rho)}, \quad A[n](\rho) \stackrel{\text{def}}{=} \rho\, W^{2r,r}[n] + \left| \eta[n](t)\, V^{2r}[n] \right|^2,$$
$$W^{2r,r}[n] \stackrel{\text{def}}{=} \left(2 \sin \frac{\pi n}{N} \right)^{2r} \frac{u^{2r}[n]}{u^{4r}[n]}, \quad V^{2r}[n] \stackrel{\text{def}}{=} \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}. \tag{11.12}$$

The minimal spline

$$S^{\rho}_{m,l}(x) = \sqrt{\frac{1}{N}} \sum_{n=-N/2}^{N/2-1} \zeta_{m,l}(\rho)[n]\, \gamma^{2r}[n]\,(t) \in {}^{2r}\mathscr{W}_{m,l}. \tag{11.13}$$

Its samples at the grid points

$$S^{\rho}_{m,l}(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \zeta_{m,l}(\rho)[n]\, V^{2r}[n]$$
$$= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \frac{\eta^*[n](t)\, \hat{\hat{z}}_{m,l}(n)}{\rho M^{2r}[n] + |\eta[n](t)|^2}$$

where

$$M^{2r}[n] \stackrel{\text{def}}{=} \frac{1}{u^{2r}[n]} \left(2 \sin \frac{\pi n}{N} \right)^{2r}, \quad V^{2r}[n] = \frac{u^{2r}[n]}{\sqrt{u^{4r}[n]}}.$$

11.1.2.3 Selection of the Regularization Parameter

Assume that we are able to evaluate the errors vector $\mathbf{e} = \{e_k\}_{k=0}^{N-1}$, $e_k = N^{-1} \sum_{n=-N/2}^{N/2-1} \omega^{kn} \hat{e}[n]$, whose variance $\mathrm{var}(\mathbf{e}) = \varepsilon^2 \approx N^{-1} \sum_{k=0}^{N-1} (e_k)^2$. Keeping in mind Eq. (11.11), denote

$$e_{m,l}[k] \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \hat{e}_{m,l}[n], \text{ where } \hat{e}_{m,l}[n] \stackrel{\text{def}}{=} \hat{e}[n]\, \frac{\left|\nu_{m,l}[n]\right|^2}{2^m},$$

$$\left(\varepsilon_{m,l}\right)^2 \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} \left(e_{m,l}[k]\right)^2 = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left|\hat{e}_{m,l}[n]\right|^2.$$

The function

$$\epsilon_{m,l}(\rho) \stackrel{\text{def}}{=} E_{m,l}\left(S^{\rho}_{m,l}\right) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left(\frac{\rho \left|\hat{\tilde{z}}_{m,l}[n]\right| M^{2r}[n]}{\rho M^{2r}[n] + |\eta[n](t)|^2} \right)^2$$

grows monotonically from zero to $N^{-1} \sum_{n=-N/2}^{N/2-1} \left|\hat{\tilde{z}}_{m,l}[n]\right|^2 = \sum_{k=0}^{N-1} \left(\tilde{z}_{m,l}[k]\right)^2$ as ρ grows from zero to infinity. Therefore, we propose to derive an optimal parameter $\bar{\rho}$ for the subspace ${}^{2r}\mathscr{W}_{m,l}$ from the equation

$$\varepsilon_{m,l}(\rho) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \left(\frac{\rho \left|\hat{\tilde{z}}_{m,l}[n]\right| M^{2r}[n]}{\rho M^{2r}[n] + |\eta[n](t)|^2} \right)^2 = \left(\varepsilon_{m,l}\right)^2. \tag{11.14}$$

11.1.2.4 Noise Modeling

We assume that the error vector $\mathbf{e}$ is a zero mean Gaussian white noise. In that case, error vector $\mathbf{e}$ can be simulated by a vector $\tilde{\mathbf{e}} = \{\tilde{e}_i\}_{i=0}^{N-1}$ in line with the scheme described in Sect. 10.3.2.1. Let $\left\{\hat{\tilde{e}}[n]\right\}_{n=-N/2}^{N/2-1}$ be the DFT spectrum of the model vector $\tilde{\mathbf{e}}$. Then, the values $\left(\varepsilon_{m,l}\right)^2$, which are needed for the parameter ρ selection, are estimated as

$$\left(\varepsilon_{m,l}\right)^2 \approx \frac{1}{2^m N} \sum_{n=-N/2}^{N/2-1} \left|(\nu_{m,l}[n])^2\, \hat{\tilde{e}}[n]\right|^2. \tag{11.15}$$

11.1.3 Approximate Inversion of Heat Equation in the Space ${}^{2r}\mathscr{S}$

11.1.3.1 Selection of the Subspaces ${}^{2r}\mathscr{W}_{m,l}$

The partial spline solution $S_{m,l}(\rho, x)$ of the inversion problem in the subspace ${}^{2r}\mathscr{W}_{m,l}$ is derived from the filtered data such that the DFT

$$\hat{\tilde{z}}_{m,l}[n] \stackrel{\text{def}}{=} \hat{z}[n]\,\frac{\left|\nu_{m,l}[n]\right|^2}{2^m}.$$

Assume that $\left\{{}^{2r}\mathscr{W}_{m,l}\right\}$ is a set of mutually orthogonal subspaces such that their union $\biguplus_{m,l} {}^{2r}\mathscr{W}_{m,l} = {}^{2r}\mathscr{S}$. Then, the complete solution of the problem can be obtained as a sum of partial solutions in the subspaces ${}^{2r}\mathscr{W}_{m,l}$. To select an optimal set of the subspaces, the relation between the spline wavelet packets $\psi^{2r}_{m,l}(t)$, whose SHA spectra are $\left\{\nu_{m,l}[n]\right\}$ and the discrete-time wavelet packets $\Psi^{2r}_{m,l}[k]$,which were defined in Sect. 9.1.3, is used. To be specific, the DFT of the wavelet packet $\Psi^{2r}_{m,l}[k]$ is $\left\{\nu_{m,l}[n]\right\}$ thus coinciding with the SHA spectrum of $\psi^{2r}_{m,l}(t)$. This relation implies the following fact.

Proposition 11.2 *There exists a one-to-one correspondence between the subspaces* ${}^{2r}\mathscr{W}_{m,l}$ *of the spline space* ${}^{2r}\mathscr{S}$ *and the subspaces* $\Pi[N]_{m,l}$ *of the signal space* $\Pi[N]$. *In addition, the SHA spectrum of a spline from* ${}^{2r}\mathscr{W}_{m,l}$ *coincides with the DFT of the respective element from* $\Pi[N]_{m,l}$.

Proof Relate the signal

$$x[k] = \sum_{\lambda=0}^{N/2^m-1} q_{m,l}[\lambda]\,\Psi^{2r}_{m,l}[k - 2^m\lambda],$$

which belongs to $Pi[N]_{m,l}$ to a spline $S(t) \in {}^{2r}\mathscr{W}_{m,l}$, which is represented as

$$S(t) = \sum_{\lambda=0}^{N/2^m-1} q_{m,l}[k]\,\psi^{2r}_{m,l}\left(t - 2^m\lambda\right).$$

Due to Eq. (8.57), the SHA spectrum of the spline $S(t)$ is $\sigma_{m,l}[n] = \nu_{m,l}[n]\,\hat{q}_{m,l}[n]_m$. On the other hand, Eq. (9.23) implies that the DFT $\hat{x}[n] = \nu_{m,l}[n]\,\hat{q}_{m,l}[n] = \sigma_{m,l}[n]$.

An implication from the proposition is that, once we determine a decomposition of the signal space $\Pi[N]$ into the set of mutually orthogonal subspaces $\Pi[N] = \biguplus_{m,l} \Pi[N]_{m,l}$, which provides an optimal split of the frequency domain of the available signal $\mathbf{z}$, it is advisable to derive partial solutions of the inversion problem in the respective subspaces ${}^{2r}\mathscr{W}_{m,l}$. It was established in Sect. 9.1.4.3 that such an optimal split of the frequency domain is achieved by construction of the best discrete-time wavelet packet basis.

Thus, selection of the relevant subspaces is implemented in the following way:

1. Apply the wavelet packet transform of order $2r$ to the data array $\mathbf{z} = \{z[k]\}$ down to the level M as was defined in Sect. 9.1. As a result, we find the coordinates $\left\{\xi_{m,l}[k]\right\}$ of the data $\mathbf{z}$ projections onto all the subspaces $\Pi[N]_{m,l},\ m = 1, \ldots, M,\ l = 0, \ldots, 2^m - 1$.

2. Apply the Best Basis algorithm (BBA), which is described in Sect. 9.1.4.3, to the transform coefficients $\{\xi_{m,l}[k]\}$. As a result, we obtain the list $ML = \{(\bar{m}, \bar{l})\}$ of indices of the discrete-time wavelet packets $\Psi_{\bar{m},\bar{l}}$, $(\bar{m}, \bar{l}) \in ML$, which generate an optimal basis for the signal $\mathbf{z}$. The list ML determines the subspaces ${}^{2r}\mathscr{W}_{\bar{m},\bar{l}}$, where partial solutions of inversion problem are derived.
3. Recall that the projection vector $\mathbf{z}_{m,l} \stackrel{\text{def}}{=} \{z_{m,l}[k]\}$ represents a noised version of the sampled spline $S_{m,l}(x,t) = \mathbf{V}_t S_{m,l}(x)$. Equations (11.6) and (11.8) imply that the DFT of the vector $\mathbf{y}_{m,l} = \left\{y_{m,l}[k] \stackrel{\text{def}}{=} S_{m,l}(k,t)\right\}$

$$\hat{y}_{m,l}[n] = \eta[n](t)\zeta_{m,l}[n]\, V^{2r}[n] = \eta[n](t)\, \nu_{m,l}[n]\, \hat{\xi}_{m,l}[n]\, V^{2r}[n],$$

where the coefficients $\eta[n](t)$ are defined in Eq. (11.3). They are exponentially decaying as n increases, therefore the vector $\mathbf{y}_{m,l}$ is effectively bandlimited. It may happen that the overlap of its DFT spectrum with some of the SHA spectra $\nu_{\check{m},\check{l}}[n]$ of the wavelet packets $\psi_{\check{m},\check{l}}(x)$ from the list ML is (almost) empty. It means that the data component $\mathbf{y}_{\check{r},\check{l}}$ contains (almost) no contribution from the function $f(x)$ that we are looking for. Thus, the projections of the signal $\mathbf{z}$ onto the subspaces ${}^{2r}\mathscr{W}_{\check{m},\check{l}}$ are (almost) pure noise. To detect such "empty" subspaces, we calculate the norms of the signals

$$\chi_{\bar{m},\bar{l}}[k] \stackrel{\text{def}}{=} \frac{1}{N}\sum_{n=-N/2}^{N/2-1} \omega^{kn}\, \eta[n](t)\, \nu_{\bar{m},\bar{l}}[n], \quad \left\|\chi_{\bar{m},\bar{l}}\right\| = \sqrt{\frac{1}{N}\sum_{n=-N/2}^{N/2-1} \left|\eta[n](t)\, \nu_{\bar{m},\bar{l}}[n]\right|^2}$$

for all pairs $(\bar{m}, \bar{l}) \in ML$ and define some threshold T. The list of subspaces ML is reduced to a shorter list $\overline{ML}$ by discarding the pairs $(\check{m}, \check{l})$ such that the norms $\left\|\chi_{\check{r},\check{l}}\right\| < T$.

11.1.3.2 Scheme for the Approximated Inversion of the Heat Equation

- Evaluate the error vector, i.e. to estimate the partial variances $\left(\varepsilon_{m,l}\right)^2$ of the noise (Eq. 11.15).
- Calculate the coefficients $\eta[n](t)$, which are defined in (Eq. 11.3).
- Implement the wavelet packet transform of order $2r$ of the signal $\mathbf{z}$ (Sect. 9.1.3, MATLAB function `spl_WP_analP.m`).
- Apply the BBA to the transform coefficients thus compiling the list ML of relevant subspaces (MATLAB function `WP_BB_Listree1D_P`).
- Reduce the list ML to $\overline{ML}$.
- Determine the optimal values $\bar{\rho}$ of the regularization parameter for each pair $(\bar{m}, \bar{l}) \in \overline{ML}$.
- Find the partial solutions $S^{\bar{\rho}}_{\bar{m},\bar{l}}(x) \in {}^{2r}\mathscr{W}_{\bar{m},\bar{l}}$ for each pair $(\bar{m}, \bar{l}) \in \overline{ML}$ (Eq. 11.13).

The approximated solution to the inversion problem **IP1** is

$$f(x) \approx S(x) = \sum_{(\tilde{m},\tilde{l})\in\overline{ML}} S^{\tilde{\rho}}_{\tilde{m},\tilde{l}}(x) \in {}^{2r}\mathscr{S}.$$

11.2 2D Block-Based Heat Equation Inversion

11.2.1 Preliminaries

In this section we outline approximate inversion of the two-dimensional heat equation

$$\begin{aligned} &\frac{\partial g(x,y,t)}{\partial t} = g''_x(x,y,t) + g''_y(x,y,t), \qquad (11.16)\\ &g(x,y,0) = f(x,y) \Longleftrightarrow g(x,y,t) = \mathbf{U}_t^2 f(x,u). \end{aligned}$$

Problem IP2 Let t be a fixed time parameter. Given $g(x,y,t) = \mathbf{U}_t^2 f(x,y)$, find $f(x,y)$.

The function $f(x,y)$, which represents the initial temperature distribution is assumed to be N-periodic in both directions. The theoretical solution of Problem IP2 is explicitly expressed in Eq. (7.21) via the 2D Fourier series. This theoretical solution is unstable due to fast decay of the Fourier coefficients. In practice, only sampled data, which, typically, is corrupted by noise, is available.

The block-based scheme of the approximate solution of Problem IP2 is a direct extension of the 1D scheme.

Assume the time parameter t is fixed and samples $\{g(k,n,t)\}$, $k,n = 0, N-1$, of the function $g(x,y,t) = \mathbf{U}_t^2$ are measured up to some zero-mean Gaussian errors $\mathbf{e} = \{\varepsilon_{k,n}\}$. Thus, the array $\mathrm{z} = \{z[k,n] = g(k,n,t) + \varepsilon_{k,n}\}$, $k,n = 0, N-1$, is available while the time moment t is known.

The function $f(x,y)$ is approximated by a spline $S(x,y) \in {}^{2r,2r}\mathscr{S}$, which is a linear combination of spline wavelet packets.

Embedding the problem into a spline setting is implemented by spline approximation of the heat equation, which is achieved in two ways.

Difference approximation: Denote, by $\mathbf{V}_t^{2d}$, the linear operator defined on the spline space ${}^{2r,2r}\mathscr{S}$ such that for $S(x,y) \in {}^{2r,2r}\mathscr{S}$, $\mathbf{V}_t^{2d} S(x,y) = s(x,y,t)$, where $s(x,y,t)$ is the spline from ${}^{2r,2r}\mathscr{S}$ (with respect to x,y), which provides a solution to the difference approximation of the heat equation

$$\frac{\partial s(x,y,t)}{\partial t} = \delta_x^2[s](x,y,t) + \delta_y^2[s](x,y,t), \quad s(x,y,0) = S(x,y).$$

Here, $\delta_x^2[q](x,y) \stackrel{\text{def}}{=} (q(x-1,y) - 2q(x,y) + q(x+1,y)$ and $\delta_y^2[q](x,y) \stackrel{\text{def}}{=} (q(x,y-1) - 2q(x,y) + q(x,y) + 1$ are the partial circular second central differences.

Collocation approximation: Denote, by $\mathbf{V}_t^{2c}$, the linear operator on ${}^{2r,2r}\mathscr{S}$ such that for a spline $S(x,y)$ from ${}^{2r,2r}\mathscr{S}$, $\mathbf{V}_t^{2c}S(x,y) = S(x,y,t)$, where $S(x,y,t)$ is the spline from ${}^{2r,2r}\mathscr{S}$, which satisfies the collocation conditions for the heat equation

$$\frac{\partial S(k,n,t)}{\partial t} = S_x''(k,n,t) + S_y''(k,n,t),\ k = 0,\ldots,N-1,\quad S(x,y,0) = S(x,y).$$

The spline $S(x,y)$ is expanded over the exponential orthonormal basis

$$S(x,y) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \sigma[\kappa,\iota]\,\gamma^{2r}[\kappa](x)\,\gamma^{2r}[\iota](y). \tag{11.17}$$

Denote, $\mathbf{V}_t^2 = \mathbf{V}_t^{2d(2c)}$. Then the splines $S(x,y,t) = \mathbf{V}_t^2 S(x,y)$ are

$$\begin{aligned}
&S(x,y,t) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \sigma(t)[\kappa,\iota]\,\gamma^{2r}[\kappa](x)\,\gamma^{2r}[\iota](y),\\
&\sigma(0)[\kappa,\iota] = \sigma[\kappa,\iota],\quad \sigma[\kappa,\iota](t) = \sigma[\kappa,\iota]\,\eta[\kappa](t)\,\eta[\iota](t),\\
&\eta[\iota](t) \stackrel{\text{def}}{=} \begin{cases} e^{-a[\iota]t}, & \text{when } \mathbf{V}_t^2 = \mathbf{V}_t^{2d}\\ e^{-b^r[\iota]t}, & \text{when } \mathbf{V}_t^2 = \mathbf{V}_t^{2c}, \end{cases}\\
&a[\iota] \stackrel{\text{def}}{=} \left(2\sin\frac{\pi\iota}{N}\right)^2,\quad b^r[\iota] \stackrel{\text{def}}{=} \left(2\sin\frac{\pi\iota}{N}\right)^2 \frac{u^{2(r-1)}[\iota]}{u^{2r}[\iota]}.
\end{aligned}$$

11.2.2 Partial Solution of the Inversion Problem in the Subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$

11.2.2.1 Splines from the Subspaces ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$

Assume a spline $S(x,y) \in {}^{2r,2r}\mathscr{S}$ is expanded over the orthonormal basis as in Eq. (11.17). The orthogonal projection of the spline $S(x,y)$ onto the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$

$$S_{m,l,\tilde{l}}(x,y) = \sum_{k,n=0}^{N_m-1} q_{m,l,\tilde{l}}[k,n]\,\psi_{m,l}^{2r}\left(x - 2^m k\right)\,\psi_{m,\tilde{l}}^{2r}\left(y - 2^m n\right), \tag{11.18}$$

where $\left\{\psi^{2r}_{m,l}\left(x(y)-2^m k(n)\right)\right\} \in {}^{2r}\mathscr{W}_{m,l(\tilde{l})} \subset {}^{2r}\mathscr{S}$ are the orthonormal wavelet packets and

$$\psi^{2r}_{m,l}\left(x-2^m k\right)\ \psi^{2r}_{m,\tilde{l}}\left(y-2^m n\right) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota]\,\omega^{-2^m(k\kappa+n\iota)}\,\gamma^{2r}[\kappa](x)\ \gamma^{2r}[\iota](y) \tag{11.19}$$

Substituting Eq. (11.19) into Eq. (11.18), we get

$$S_{m,l,\tilde{l}}(x,y) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \zeta_{m,l,\tilde{l}}[\kappa,\iota]\,\gamma^{2r}[\kappa](x)\ \gamma^{2r}[\iota](y), \tag{11.20}$$

$$\zeta_{m,l,\tilde{l}}[\kappa,\iota] \stackrel{\text{def}}{=} \nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota]\,\hat{q}_{m,l,\tilde{l}}[\kappa,\iota]_m.$$

The DFT sequence $\hat{q}_{m,l,\tilde{l}}[\kappa,\iota]_m = \sum_{k,n=0}^{N_m-1}\omega^{-2^m(k\kappa+n\iota)}\,q_{m,l,\tilde{l}}[k,n]$ is N_m-periodic in both directions, where $N_m = N/2^m$.

Then, the spline $S_{m,l,\tilde{l}}(x,y,t) = \mathbf{V}^2_t\,S_{m,l,\tilde{l}}(x,y)$ is

$$S_{m,l,\tilde{l}}(x,y,t) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \zeta_{m,l,\tilde{l}}[\kappa,\iota](t)\,\gamma^{2r}[\kappa](x)\ \gamma^{2r}[\iota](y),$$

$$\zeta_{m,l,\tilde{l}}[\kappa,\iota](t) = \zeta_{m,l,\tilde{l}}[\kappa,\iota]\,\eta[\kappa](t)\,\eta[\iota](t),$$

where $\eta[\iota](t)$ are defined in Eq. (11.18).

Similarly to 1D case (Proposition 11.1), the following approximate representation of the projection coordinates holds

$$\zeta_{m,l,\tilde{l}}[\kappa,\iota]] \approx \frac{1}{4^m}\,\sigma[\kappa,\iota]\,\left|\nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota]\right|^2. \tag{11.21}$$

Denote, by $Y[k,n] \stackrel{\text{def}}{=} S(k,n,t)$, samples of the spline $S(x,y,t) = \mathbf{V}^2_t\,S(x,y)$:

$$Y[k,n] = \sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{2^m(k\kappa+n\iota)}\,\sigma[\kappa,\iota]\ \eta[\kappa](t)\,\eta[\iota](t)\,V^{2r}[\kappa]\,V^{2r}[\iota]$$

$$\hat{Y}[\kappa,\iota] = \sigma[\kappa,\iota]\ \eta[\kappa](t)\,\eta[\iota](t)\,V^{2r}[\kappa]\,V^{2r}[\iota], \quad V^{2r}[\iota] = \frac{u^{2r}[\iota]}{\sqrt{u^{4r}[\iota]}}.$$

Then, the samples of the spline $S_{m,l,\tilde{l}}(x, y, t) = \mathbf{V}_t^2\, S_{m,l,\tilde{l}}(x, y)$ are

$$Y_{m,l,\tilde{l}}[k, n] \stackrel{\text{def}}{=} S_{m,l,\tilde{l}}(k, n, t) = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{2^m(k\kappa+n\,\iota)}\, \zeta_{m,l,\tilde{l}}[\kappa, \iota]\, \eta[\kappa](t)\, \eta[\iota](t)\, V^{2r}[\kappa] \tag{11.22}$$

The following relation is derived from Eq. (11.21) similarly to Eq.(11.9):

$$\hat{Y}_{m,l,\tilde{l}}[\kappa, \iota] = \zeta_{m,l,\tilde{l}}[\kappa, \iota]\, \eta[\kappa](t)\, \eta[\iota](t)\, V^{2r}[\kappa] \approx \frac{1}{4^m} \hat{Y}[\kappa, \iota] \left|\nu_{m,l}[\kappa]\, \nu_{m,\tilde{l}}[\iota]\right|^2 \tag{11.23}$$

Remark 11.2.1 Equation (11.23) can be interpreted in a sense that confinement of the operator's $\mathbf{V}_t^2$ domain from the whole spline space ${}^{2r,2r}\mathscr{S}$ to the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$, effectively results in multiplication of the 2D DFT $\hat{Y}[\kappa, \iota]$ of the sampled output $Y[k, n] = \mathbf{V}_t^2 S_{m,l,\tilde{l}}(k, n)$, $k, n = 0, \ldots, N-1$, with the factor $4^{-m} \left|\nu_{m,l}[\kappa]\, \nu_{m,\tilde{l}}[\iota]\right|^2$.

11.2.2.2 Parameterized Spline Solution in the Subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$

The scheme of partial solution of Problem IP2 in the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ is similar to the scheme for Problem IP1, which is presented in Sect. 11.1. By assumption, t is a known time parameter and the array $\mathbf{z} = \left\{z[k, n] = g(k, n, t) + e_{k,n}\right\} = \mathbf{g} + \mathbf{e}$ is available, where $g(x, y, t) = \mathbf{U}_t^2\, f(x, y)$. We seek for a partial approximate inversion of the two-dimensional heat Eq. (11.16). Keeping in mind Remark 11.2.1, prepare the "filtered" data array

$$\tilde{z}_{m,l,\tilde{l}}[k, n] \stackrel{\text{def}}{=} \frac{1}{N^2} \sum_{\kappa,ii=-N/2}^{N/2-1} \omega^{(k\kappa+n\,\iota)}\, \hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa, \iota],$$

where

$$\hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa, \iota] \stackrel{\text{def}}{=} \hat{z}[\kappa, \iota]\, \frac{\left|\nu_{m,l}[\kappa]\, \nu_{m,\tilde{l}}[\iota]\right|^2}{4^m}, \quad \hat{z}[\kappa, \iota] = \sum_{k,n=0}^{N-1} \omega^{-(k\kappa+n\,\iota)}\, z[k, n].$$

The solution is derived as a spline $S_\rho(x, y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}$, which minimizes the functional $J_{m,l,\tilde{l}}(\rho)(S) \stackrel{\text{def}}{=} \rho I(S) + E_{m,l,\tilde{l}}(S)$, where ρ is a numerical parameter,

$$I(S) \stackrel{\text{def}}{=} \|S_x^{(r)}\|^2 + \|S_y^{(r)}\|^2, \quad (E_{m,l,\tilde{l}}(S) \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1} \Big(S(k,n,t) - \tilde{z}_{m,l,\tilde{l}}[k,n]\Big)^2,$$

$S(x,y,t) \stackrel{\text{def}}{=} \mathbf{V}_t^2 S(x,y)$ and $\mathbf{V}_t^2 = \mathbf{V}_t^{2d(2c)}$.

Assume that a spline $s(x,y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ is represented as in Eq. (11.20)

$$s(x,y) = \frac{1}{N} \sum_{\kappa,\iota=-N/2}^{N/2-1} \zeta_{m,l,\tilde{l}}[\kappa,\iota]\, \gamma^{2r}[\kappa](x)\ \gamma^{2r}[\iota](y).$$

Then, Eq. (4.63) implies that

$$I(s) = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \Big(W^{2r,r}[\kappa] + W^{2r,r}[\iota]\Big)\ |\zeta_{m,l,\tilde{l}}[\kappa,\iota]|^2,$$

$$W^{2r,r}[\iota] \stackrel{\text{def}}{=} \Big(2 \sin\frac{\pi\iota}{N}\Big)^{2r} \frac{u^{2r}[\iota]}{u^{4r}[\iota]}.$$

The discrepancy functional

$$E_{m,l,\tilde{l}}(s) = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \Big|\eta[\kappa](t)\,\eta[\iota](t)\,\zeta_{m,l,\tilde{l}}[\kappa,\iota]\,V^{2r}[\kappa]\,V^{2r}[\iota] - \hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa,\iota]\Big|^2,$$

where $V^{2r}[\iota] = u^{2r}[\iota]/\sqrt{u^{4r}[\iota]}$. Consequently,

$$J_{m,l,\tilde{l}}(\rho)(s) = \frac{1}{N^2} \sum_{\kappa,\iota=-N/2}^{N/2-1} \rho\,|\zeta_{m,l,\tilde{l}}[\kappa,\iota]|^2 \Big(W^{2r,r}[\kappa] + W^{2r,r}[\iota]\Big) + \Big|\eta[\kappa](t)\,\eta[\iota](t)\,\zeta_{m,l,\tilde{l}}[\kappa,\iota]\,V^{2r}[\kappa]\,V^{2r}[\iota] - \hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa,\iota]\Big|^2.$$

The functional $J_{m,l,\tilde{l}}(\rho)(s)$ achieves its minimum when

$$\zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota] = \frac{\eta^*[\kappa](t)\ \eta^*[\iota](t)\ V^{2r}[\kappa]\ V^{2r}[\iota]\,\hat{\tilde{z}}_{m,l,\tilde{l}}(n)}{A[\kappa,\iota](\rho)},$$

$$A[\kappa,\iota](\rho) \stackrel{\text{def}}{=} \rho\,\Big(W^{2r,r}[\kappa] + W^{2r,r}[\iota]\Big) + \Big|\eta[\kappa](t)\,\eta[\iota](t)\ V^{2r}[\kappa]\ V^{2r}[\iota]\Big|^2,$$

$$W^{2r,r}[\iota] \stackrel{\text{def}}{=} \Big(2 \sin\frac{\pi\iota}{N}\Big)^{2r} \frac{u^{2r}[\iota]}{u^{4r}[\iota]}, \quad V^{2r}[\iota] \stackrel{\text{def}}{=} \frac{u^{2r}[\iota]}{\sqrt{u^{4r}[\iota]}}.$$

The minimal spline

$$S^{\rho}_{m,l,\tilde{l}}(x,y) = \frac{1}{N}\sum_{\kappa,\iota=-N/2}^{N/2-1} \zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota]\ \gamma^{2r}[\kappa](x)\,\gamma^{2r}[\iota](y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}}.$$

Its samples at the grid points are:

$$\begin{aligned} S^{\rho}_{m,l,\tilde{l}}(k,n) &= \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(k\kappa+n\,\iota)}\zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota]\ V^{2r}[\kappa]\ V^{2r}[\iota] \\ &= \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(k\kappa+n\,\iota)}\ \frac{\eta^*[\kappa](t)\ \eta^*[\iota](t)\,\hat{\hat{z}}_{m,l,\tilde{l}}}{\rho\left(M^{2r}[\kappa]+M^{2r}[\iota]\right)+|\eta[\kappa](t)\ \eta[\iota](t)|^2} \end{aligned}$$

where

$$M^{2r}[\iota] \stackrel{\text{def}}{=} \frac{1}{u^{2r}[\iota]}\left(2\,\sin\frac{\pi\iota}{N}\right)^{2r}, \quad V^{2r}[\iota] = \frac{u^{2r}[\iota]}{\sqrt{u^{4r}[\iota]}}.$$

11.2.2.3 Selection of the Regularization Parameter

The errors array

$$\mathbf{e} = \left\{e_{k,n}\right\}_{k,n=0}^{N-1}, \quad e_{k,n} = \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(k\kappa+n\,\iota)}\hat{e}[\kappa,\iota],$$

is evaluated similarly to 1D case. Denote,

$$\begin{aligned} &e_{m,l,\tilde{l}}[k,n] \stackrel{\text{def}}{=} \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1} \omega^{(k\kappa+n\,\iota)}\hat{e}_{m,l,\tilde{l}}[\kappa,\iota], \text{ where} \\ &\hat{e}_{m,l,\tilde{l}}[\kappa,\iota] \stackrel{\text{def}}{=} \hat{e}[\kappa,\iota]\ \frac{\left|\nu_{m,l}[\kappa]\,\nu_{m,\tilde{l}}[\iota]\right|^2}{4^m}, \\ &\left(\varepsilon_{m,l,\tilde{l}}\right)^2 \stackrel{\text{def}}{=} \sum_{k,n=0}^{N-1}\left(e_{m,l,\tilde{l}}[k,n]\right)^2 = \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\left|\hat{e}_{m,l,\tilde{l}}[\kappa,\iota]\right|^2. \end{aligned}$$

The optimal parameter $\bar{\rho}$ for the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ is derived from the equation

$$\varepsilon_{m,l,\tilde{l}}(\rho) \stackrel{\text{def}}{=} E_{m,l,\tilde{l}}\left(S^{\rho}_{m,l,\tilde{l}}(\cdot,\cdot)\right) \tag{11.24}$$
$$= \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\left(\frac{\rho\left|\hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa,\iota]\right|\,\left(M^{2r}[\kappa]+M^{2r}[\iota]\right)}{\rho\left(M^{2r}[\kappa]+M^{2r}[\iota]\right)+|\eta[\kappa](t)\,\eta[\iota](t)|^2}\right)^2 = \left(\varepsilon_{m,l,\tilde{l}}\right)^2.$$

11.2.3 Scheme for the Approximate Inversion of the 2D Heat Equation

11.2.3.1 Best Basis Algorithm (BBA)

This method is a direct extension of the 1D algorithm, which is described in Sect. 10.1.1. Assume the time parameter t is fixed and the data array

$$\mathrm{z} = \left\{z[k,n] = g(k,n,t) + \varepsilon_{k,n}\right\},\ k,n = 0, N-1,$$

is available, where $g(x,y,t) = \mathbf{U}_t^2 f(x,y)$. The algorithm is implemented by the following steps.

1. Model the error array $\mathbf{e}$.
2. Implement the 2D wavelet packet transform of order $2r$ of the data $\mathbf{z}$.
3. Apply the BBA, which is described in Sect. 10.2.3, to the transform coefficients. As a result, obtain the list $ML\tilde{L} = \left\{(\bar{m},\bar{l},\bar{\tilde{l}})\right\}$ such that the shifts of the discrete-time wavelet packets $\Psi_{\bar{m},\bar{l},\bar{\tilde{l}}}$, where $(\bar{m},\bar{l},\bar{\tilde{l}}) \in ML\tilde{L}$, form an optimal basis for the array $\mathbf{z}$. The list $ML\tilde{L}$ determines the subspaces ${}^{2r}\mathscr{W}^{\bar{m},\bar{l},\bar{\tilde{l}}}$, where the partial solutions to Problem IP2 are to be derived.
4. To detect "empty" subspaces, calculate the norms of the signals

$$\chi_{\bar{m},\bar{l},\bar{\tilde{l}}}[k] \stackrel{\text{def}}{=} \frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(k\kappa+n\,\iota)}\,\eta[\kappa](t)\,\eta[\iota](t)\,\nu_{\bar{m},\bar{l}}[\kappa]\,\nu_{\bar{m},\bar{\tilde{l}}}[\iota],$$

$$\left\|\chi_{\bar{m},\bar{l},\bar{\tilde{l}}}\right\| = \frac{1}{N}\sqrt{\sum_{\kappa,\iota=-N/2}^{N/2-1}\left|\eta[\kappa](t)\,\eta[\iota](t)\,\nu_{\bar{m},\bar{l}}[\kappa]\,\nu_{\bar{m},\bar{\tilde{l}}}[\iota]\right|}$$

for all triples $(\bar{m},\bar{l},\bar{\tilde{l}}) \in ML\tilde{L}$.

5. Define some threshold T. The list of subspaces $MLL\tilde{L}$ is reduced to a shorter lisT $\overline{MLL\tilde{L}}$ by discarding the triples $(\check{m}, \check{l}, \check{\tilde{l}})$ such that the norms $\left\| \chi_{\check{r},\check{l},\check{\tilde{l}}} \right\| < T$.
6. Determine the optimal values $\bar{\rho}$ of the parameter for each triple $(\bar{r}, \bar{l}, \bar{\tilde{l}}) \in \overline{MLL\tilde{L}}$.
7. Find the partial solutions $S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x, y) \in {}^{2r}\mathscr{W}_{\bar{m},\bar{l},\bar{\tilde{l}}}$ for each triple $(\bar{r}, \bar{l}, \bar{\tilde{l}}) \in \overline{MLL\tilde{L}}$.
8. The approximated solution to Problem IP2 is

$$S(x, y) = \sum_{\bar{m},\bar{l},\bar{\tilde{l}}} S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x, y) \in {}^{2r,2r}\mathscr{S}.$$

11.2.3.2 A Variation of BBA: Block Pursuit Algorithm (BPA)

This method is, to some extent, similar to the Matching Pursuit method presented in Chap. 11

1. Evaluate the error array **e**.
2. Set a threshold T for the stopping rule.
3. Set a number B of best blocks.
4. Put $\mathbf{z}^0 = \mathbf{z}$ and $S^0(x, y) \equiv 0$.
5. Implement the 2D wavelet packet transform of order $2r$ of the data $\mathbf{z}^0$.
6. Apply the BBA to the transform coefficients thus obtaining the list $MLL\tilde{L} = \left\{(\bar{m}, \bar{l}, \bar{\tilde{l}})\right\}$.
7. Calculate normalized energies $E_{\bar{m},\bar{l},\bar{\tilde{l}}}$ (the l_2 norms divided by the number of coefficients in the block) for each coefficients' block from the list $MLL\tilde{L}$ and sort them in the descending order.
8. Select B subspaces $\Pi_{\bar{m},\bar{l},\bar{\tilde{l}}}$ whose indices occupy first B places in the sorted list, thus obtaining the list $MLL\tilde{L}_B$ with B entries.
9. Discard from the list $MLL\tilde{L}_B$ the entries, whose energies

$$E_{\bar{m},\bar{l},\bar{\tilde{l}}} < T \tag{11.25}$$

   (if such entries are present).
10. If the inequality (11.25) holds for all the entries in the list $MLL\tilde{L}_B$ then stop the algorithm.

    Otherwise:
11. Determine the optimal values $\bar{\rho}$ of the parameter for the subspaces ${}^{2r}\mathscr{W}_{\bar{m},\bar{l},\bar{\tilde{l}}}$ from the (possibly reduced) list $MLL\tilde{L}_B$.
12. Find the partial solutions $S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x, y) \in {}^{2r}\mathscr{W}_{\bar{m},\bar{l},\bar{\tilde{l}}}$.
13. Put $S^1(x, y) \stackrel{\text{def}}{=} S^0(x, y) + \sum_{\bar{m},\bar{l},\bar{\tilde{l}} \in MLL\tilde{L}_B} S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x, y)$.

14. Using Eq. (11.22), derive the arrays $\left\{Y_{\bar{m},\bar{l},\bar{\tilde{l}}}[k,n]\right\}$, $k,n,=0,\ldots,N-1$, which are the sampled splines $S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x,y,t) \stackrel{\text{def}}{=} \mathbf{V}_t^2 S^{\bar{\rho}}_{\bar{m},\bar{l},\bar{\tilde{l}}}(x,y)$.
15. Put $\mathbf{z}^1 \stackrel{\text{def}}{=} \left\{z^0[k,n] - \sum_{\bar{m},\bar{l},\bar{\tilde{l}}\in MLL_B} Y_{\bar{m},\bar{l},\bar{\tilde{l}}}[k,n]\right\}$.
16. Iterate the procedures, starting from Item 5 using $\mathbf{z}^1$ instead of $\mathbf{z}^0$.

Output: The approximated solution to Problem IP2 is

$$S(x,y) = \sum_i S^i(x,y) \in {}^{2r,2r}\mathscr{S}.$$

11.2.3.3 Comments

1. The described algorithms can be utilized for signals' and images' denoising when the time parameter $t = 0$. In this case, the general scheme remains unchanged. The approximated solutions produced by the global Tikhonov algorithm (GTA), which were described in Sect. 7.2, become the smoothing one- or two-dimensional splines.
2. The BPA can be utilized in two modes:

   Quasi-Matching Pursuit: It uses a small number (may be just one) of the Best Basis blocks per iteration and makes a large number of iterations thus successively approaching the solution.

   Update of the BBA: It uses all the blocks contained in the reduced Best Basis list and makes 2 or 3 iterations thus updating the result from the BBA.

   It is more computationally expensive compared to BBA but, typically, produces more accurate results.
3. The block-based algorithms for inversion of the heat equations, which are presented in this chapter are applicable, with some obvious modifications, to solving the deconvolution problem. A detailed presentation of the block-based deconvolution is given in [2].

Outline of the block-based deconvolution

In 2D case, the partial approximate solution of the convolution equation Eq. (7.11) in the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ is presented by the spline

$$S^{\rho}_{m,l,\tilde{l}}(x,y) = \frac{1}{N} \sum_{\kappa,\iota=-N/2}^{N/2-1} \zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota]\, \gamma^{2r}[\kappa](x)\, \gamma^{2r}[\iota](y) \in {}^{2r}\mathscr{W}_{m,l,\tilde{l}},$$

with the coefficients

$$\zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota] = \frac{\hat{h}[\kappa,\iota]^*\, T^{2r,q}[\kappa]\, T^{2r,q}[\iota]\, \hat{\tilde{z}}_{m,l,\tilde{l}}(n)}{A[\kappa,\iota](\rho)},$$

$$A[\kappa,\iota](\rho) \stackrel{\text{def}}{=} \rho\left(W^{2r,r}[\kappa]+W^{2r,r}[\iota]\right)+\left|\hat{h}[\kappa,\iota]\ T^{2r,q}[\kappa]\ T^{2r,q}[\iota]\right|^2,$$

$$W^{2r}[\iota] \stackrel{\text{def}}{=} \left(2\ \sin\frac{\pi\iota}{N}\right)^{2r}\ \frac{u^{2r}[\iota]}{u^{4r}[\iota]},\quad T^{2r,q}[\iota] \stackrel{\text{def}}{=} \frac{u^{2r+q}[\iota]}{\sqrt{u^{4r}[\iota]}\,u^{q}[n]}.$$

Its samples at the grid points are given by

$$S^{\rho}_{m,l,\tilde{l}}(k,n)=\frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\omega^{(k\kappa+n\,\iota)}\zeta_{m,l,\tilde{l}}(\rho)[\kappa,\iota]\ V^{2r}[\kappa]\ V^{2r}[\iota],$$

where $V^{2r}[\iota]=u^{2r}[\iota]/\sqrt{u^{4r}[\iota]}$. The optimal parameter $\bar{\rho}$ for the subspace ${}^{2r}\mathscr{W}_{m,l,\tilde{l}}$ is derived from the equation

$$\frac{1}{N^2}\sum_{\kappa,\iota=-N/2}^{N/2-1}\left(\frac{\rho\left|\hat{\tilde{z}}_{m,l,\tilde{l}}[\kappa,\iota]\right|\ \left(W^{2r,r}[\kappa]+W^{2r,r}[\iota]\right)}{A[\kappa,\iota](\rho)}\right)^2=\left(\varepsilon_{m,l,\tilde{l}}\right)^2.$$

11.3 Numerical Examples

The following are examples derived from three groups of experiments on using the block-based methods for restoration of 2D images:

Denoising: Restoration of objects corrupted by Gaussian noise (the time parameter $t=0$).

Clean blurred input: Restoration of blurred objects when the time parameter $t>0$. In this case, although noise is not introduced, some errors of measurements are present. Therefore, the problem remains ill-posed. The advantage of the block-based methods over the global ones consists in accurate tuning the frequency bands, where the solution is looked for, to effective frequency domain of the blurred image. The BPA does this more precisely than the BBA due to wider choice of available subspaces.

Noised blurred input: Restoration of objects from blurred inputs, which are corrupted by Gaussian noise.

The examples illustrate the difference between the performance of the GTA presented in Sect. 7.1, of the BBA and of the BPA. Visual perception is compared and the peak signal to noise ratio (PSNR) in decibels (Eq. 6.42).

All the experiments are performed using the MATLAB code `Block_runP`, which implements the block-based inversion of the heat equation and the deconvolution in an interactive mode and compares the results with the results from the GTA.

Three benchmark images each of which is presented by 512×512 array of samples are used as the initial temperature distributions (Fig. 11.2).

Fig. 11.2 *Top left* "Barbara". *Top right* "Lena". *Bottom* "Fingerprint"

Example 1 Barbara denoising

The "Barbara" image, which was corrupted by Gaussian zero-mean noise with standard deviations STD=25 serves as an input. The time parameter $t = 0$. Figure 11.3 displays fragments of the noised input image and of the image restored by GTA, BBA and BPA. One can observe that BPA produces the highest PSNR value. The noise is satisfactorily suppressed while the fine texture is restored. The BBA result is very similar to that of the BPA. The GTA performance in the texture restoration is inferior compared to either the performance of the BBA or that of the BPA. The results are displayed in Fig. 11.3.

Example 2 Restoration of the strongly blurred "Fingerprint" image

In this example, the "Fingerprint" image was used as an initial temperature distribution. The input presents the temperature distribution when the time parameter $t = 46$.

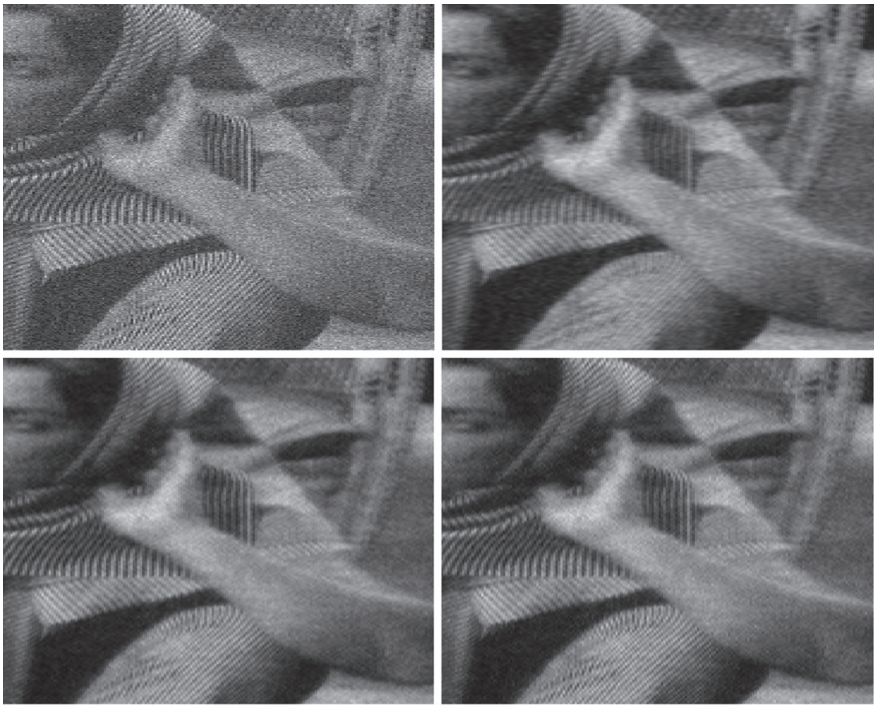


Fig. 11.3 "Barbara". *Top left* noised image, STD = 25, PSNR = 20.16; *Top right* Image restored by GTA, PSNR = 24.68. *Bottom left* Image restored by BBA, PSNR = 2 5.34, spline wavelet packets of tenth order from 2 levels were used. *Bottom right* Image restored by BPA, PSNR = 25.44, spline wavelet packets of tenth order from 2 levels were used. Number of best blocks per iteration $B = 4$. The result is achieved by 4 iterations

The best result is achieved by the BPA method although with a relatively high computational cost (1000 iterations were implemented, which took about 5 minutes). The BPA restored the texture of the "Fingerprint" image, which was completely smeared in the input. The output quality from the BBA is slightly inferior to that of the BPA. The GTA produced a much worse result. The results are displayed in Fig. 11.4.

Example 3 Deconvolution of blurred "Barbara"

This example illustrates application of block-based deconvolution algorithms to the restoration of the "Barbara", image, which was convolved with the Gaussian kernel, STD=5. As in the "Fingerprint" example, the best result is achieved by the BPA method. The BPA restored the texture of the image, which was smeared in the input. The quality of the output from the BBA is almost the same as from the BPA but it is achieved by a much lower computational cost. The GTA produced a worse result. The results are displayed in Fig. 11.5.

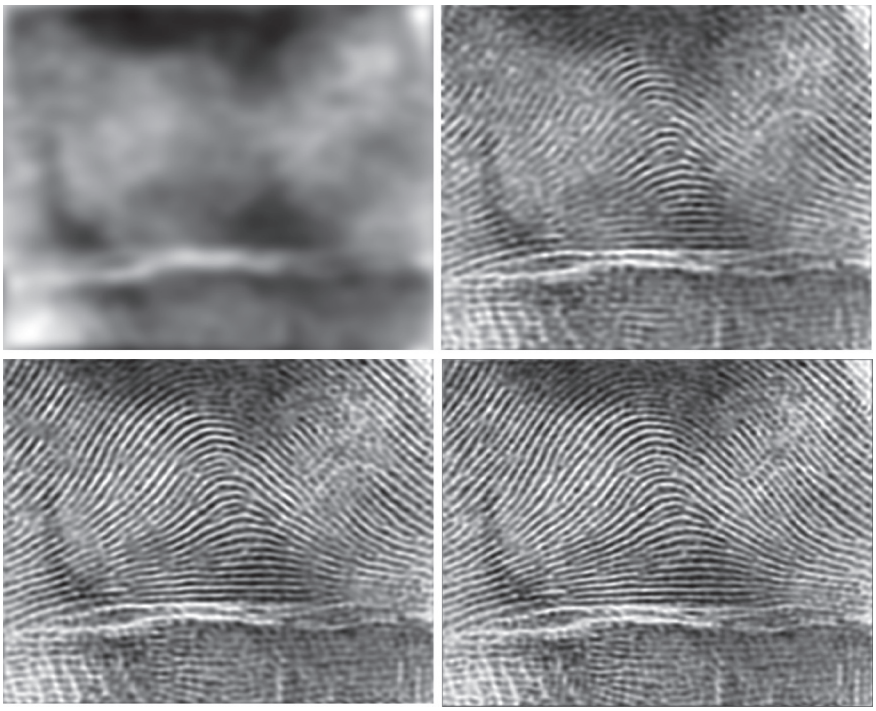


Fig. 11.4 "Fingerprint". *Top left* Blurred image, t = 46, PSNR = 15.71; *Top right* Image restored by GTA, PSNR = 19. *Bottom left* Image restored by BBA, PSNR = 21.22, spline wavelet packets of tenth order from 3 levels were used. *Bottom right* Image restored by BPA, PSNR = 22.89, spline wavelet packets of tenth order from 2 levels were used. Number of best blocks per iteration $B = 5$. The result is achieved by 1000 iterations

Example 4 Restoration of blurred and noised "Lena"

In this example, the "Lena" image was used as an initial temperature distribution. The input presents the temperature distribution when the time parameter $t = 2$. The blurred image is corrupted by Gaussian noise, whose STD=10. The best result is achieved by the BPA method. The BPA-restored image is sharper compared to two other images, and its PSNR is the highest. Noise is successfully suppressed in all the restored images. The results are displayed in Fig. 11.6. In this example the BPA acted in the update mode: it used all the blocks involved in the BBA and made only two iterations. The first iteration produced the same result as the BBA, while the second one updated it.

Fig. 11.5 'Barbara'. *Top left* Image convolved with the Gaussian kernel, STD = 5, PSNR = 21.71; *Top right* Image restored by GTA, PSNR = 24.31. *Bottom left* Image restored by BBA, PSNR = 24.55, spline wavelet packets of fourth order from 4 levels were used. *Bottom right*: Image restored by BPA, PSNR = 24.65, spline wavelet packets of twelfth order from one level were used. Number of best blocks per iteration $B = 1$. The result is achieved by 200 iterations

Comments

The block-based algorithms described in this chapter provide stable approximated solutions to such ill-posed problems as inversion of the heat equation and deconvolution of signals and images where available data are discrete and noised. The solutions are provided by spline functions, which are linear combinations of orthonormal wavelet packets. Diversity of spline wavelet packets enables us to have flexible adaptation of the algorithm to the available data satisfying desirable properties of the solution. The adaptation is achieved automatically. The only a-priori information needed is noise evaluation, which is discussed in Sect. 11.1.2.4. The SHA technique, which perfectly fits to convolution problems, yields powerful tools for fast implementation of the algorithm.

The basic idea of the method is to implement the regularized inversion (deconvolution) of signals (images), separately in different frequency bands by assuming that the relative portions of the coherent signal and noise are different in different

Fig. 11.6 "Lena". *Top left* Blurred noised image, t = 2, noise STD = 10, PSNR = 25.15; *Top right* Image restored by GTA, PSNR = 27.93. *Bottom left* Image restored by BBA, PSNR = 28.20, spline wavelet packets of the fourth order from 6 levels were used, 600 waveforms were involved. *Bottom right* Image restored by BPA, PSNR = 28.29, spline wavelet packets of the fourth order from 6 level were used. Number of best blocks per iteration $B = 600$. The result is achieved by 2 iterations

frequency bands. Representation of an input signal (image) in a variety of frequency bands is achieved by its expansion with orthonormal bases formed from translations of the spline wavelet packets. The spectra of these wavelet packets are close to rectangular and form a variety of splits of the signal's frequency domain. An optimal split is achieved by using the Best Basis scheme (BBA). Additional adaptation abilities stem from varying orders of the involved splines and the depths of the decomposition. Another adaptation scheme is implemented via the so-called BPA, which can be implemented either in a quasi—Matching Pursuit mode or in a mode, which updates the BBA result.

The conducted experiments prove the efficiency of the algorithm for solving the problem of inversion of the heat equation/deconvolutionfrom the discrete noised data. In addition, the algorithm was efficient in the two extreme cases: 1. Denoising an object, which was not blurred. 2. Deblurring an object when noise is not present. Since the presented methods extend the classical Tikhonov regularization method, their performance was compared with the performance of the global spline-based Tikhonov algorithm (GTA). In all the experiments, the BBA and, especially, the BPA produced better visual quality and higher PSNR than the GTA. That was achieved at the expense of heavier computational load. While processing of a 512×512

image is carried out by the GTA in a fraction of second, the BBA requires a couple of seconds, On the other hand, BPA processing may take from a couple of seconds to minutes (depending on the kernel's passband and depth of decomposition).

The library of spline wavelet packets supplied with an efficient implementation scheme can serve as a tool in many other applied problems where an adaptive split of the frequency domain is needed. One such application is acoustic pattern recognition [1, 3].

References

1. A. Averbuch, E. Hulata, V. Zheludev, I. Kozlov, A wavelet packet algorithm for classification and detection of moving vehicles. Multidimension. Syst. Signal Process. **12**(1), 9–31 (2001)
2. A. Averbuch, V. Zheludev, P. Neittaanmäki, J. Koren, Block based deconvolution algorithm using spline wavelet packets. J. Math. Imaging Vision **38**(3), 197–225 (2010)
3. A. Averbuch, V. Zheludev, N. Rabin, A. Schclar, Wavelet-based acoustic detection of moving vehicles. Multidimension. Syst. Signal Process. **20**(1), 55–80 (2009)
4. R.R. Coifman, V.M. Wickerhauser, Entropy-based algorithms for best basis selection. IEEE Trans. Inf. Theory **38**(2), 713–718 (1992)
5. S. Mallat, Z. Zhang, Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. **41**(12), 3397–3415 (1993)
6. M.V. Wickerhauser, *Adapted Wavelet Analysis: From Theory to Software* (AK Peters, Wellesley, 1994)

Chapter 12
Hydro-Acoustic Target Detection

Abstract This chapter presents an example of utilization of the discrete–time wavelet packets, which are described in Sect. 9.1, to classification of acoustic signals and detection of a target. The methodology based on wavelet packets is applied to a problem of detection of a boat of a certain type when other background noises are present. The solution is obtained via analysis of boat's hydro-acoustic signature against an existing database of recorded and processed hydro-acoustic signals. The signals are characterized by the distribution of their energies among blocks of wavelet packet coefficients.

This chapter describes acoustic detection of a certain boat while the background conditions are highly variable. Energies in different blocks of wavelet packet coefficients are used as characteristic features of the acoustic signals. To derive the acoustic signature of the boat of interest, a modification of the the Best Discriminant Basis method [11] is used. The decision is made by combining answers from the following classifiers: Linear Discriminant Analysis (LDA) [8], Nearest Neighbor (NN) classifier and the Classification and Regression Trees (CART) [6] that is also supplied with an additional unit called *Aisles* that reduces false alarm. The presented algorithm provides a generic solution for many classification and detection problems, which deal with signals, whose structure is close to periodic[1–5]. The presentation is based on the paper [4].

12.1 Outline of the Problem

The goal is to detect, with minimal number of false alarms and miss-detections, arrivals of boats of a certain type via analysis of their hydro-acoustic signatures. This processing is done against an existing database of recorded hydro-acoustic signals. The problem is complicated because many types of boats arrive at the scene. In addition, the following may constantly change: surrounding conditions, velocities and directions of the boats of interest, and their distances from the receiver, not to name some affecting conditions of the recorded acoustics.

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_12,

Fig. 12.1 The blue tourist boat (**BTB**)

As a running example, detection of the blue tourist boat (denoted **BTB**) that is seen in Fig. 12.1 is to be described. A successful detection depends on deriving an acoustic signature of the target that is built from characteristic features. This signature enables us to discriminate between the target of interest and background objects.

Hydro-acoustics signals emitted by boats have quasi-periodic structure. This is due mainly to the sounds emitted by their engines and propellers. Each acoustic signal contains only a few dominating bands in the frequency domain. As the boat moves, conditions are changing and configuration of these bands may vary, but their general disposition remains. Therefore, we assume that the acoustic signature for the class of signals emitted by a certain boat is obtained as a combination of the inherent energies in the blocks of the wavelet packet coefficients of the signals, each of which is related to a certain frequency band, as was discussed in Sect. 9.1.4. Thus, they can be used as candidates to become selected features.

The experiments demonstrate that blocks of wavelet packet coefficients provide distinctive characteristic features, which accurately discriminate between the *Boats* (**B**) and the *NoBoats* (**NB**) classes and between the *BlueTouristBoat* (**BTB**) and *OtherBoat* (**OB**) classes. Extracting the characteristic features (parameters) and deriving the acoustic signatures for these classes are critical tasks in the training phase of the process.

For identification of a boat by its acoustic signatures, the final phase of the detection process combines outputs from three classifiers, which are: 1. Linear Discriminant Analysis (LDA) classifier [8]. 2. Classification and Regression Trees (CART) classifier [6]. 3. CART supplied with an additional unit called *Aisles* that reduces false alarms.

Fig. 12.2 Examples of other boats: the commuter boat, the fire boat and the ferry ship

12.2 Structure of the Recorded Data

The audio recordings were acquired by a single hydrophone. For reference, these recordings were accompanied by simultaneous video footage. The recordings were taken under different conditions. Some recordings were made when no boats were present at the scene. Some recorded the **BTB**, which was cruising at different speeds, directions and distances from the hydrophone. From time to time, other boats of different types, such as fire boats, commuter boats, big ferry ships, and small motor boats, were present at the scene together with the **BTB**. Figure 12.2 displays some vessels, which arrived at the scene.

The audio recordings were sampled at the rate of 2000 samples per second. Part of the recordings was used for training the algorithm while the remaining recordings were left for testing.

Fragments of training recordings that did not contain boats sounds were stored as **NB**-class signals. Fragments containing sounds emitted by various boats were stored as **B**-class signals. The signals in the latter class were highly variable. This class was divided into two subclasses: 1. Fragments containing recordings of the Blue

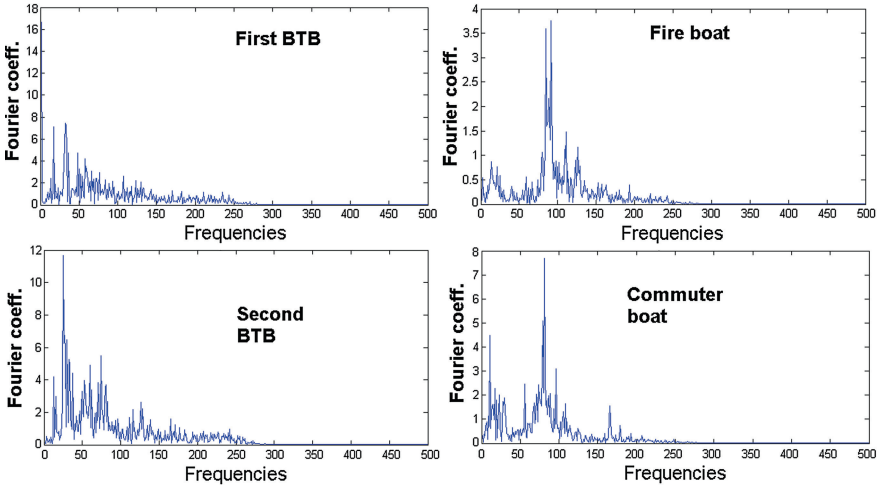


Fig. 12.3 Fourier spectra of fragments from different recordings. *Left* **BTB** in different positions. *Right* two other boats

Tourist Boat, were designated as **BTB**-class. 2. The rest of the **B**-class signals were designated as **OB**-class.

Examination of the Fourier spectra of the available signals led to the showed that spectra of the signals differ significantly from each other under different cruising conditions, even within the same **BTB** class. However, there are some common properties to all the **BTB** acoustic signals. These signals are quasi-periodic in the sense that there exist some dominating frequencies in each signal. These frequencies may vary as cruising conditions change. However, for the same boat, these variations are confined in narrow frequency bands. Moreover, relative locations of the frequency bands are stable (invariant), to some extent, for signals that are emitted by the same boat. We illustrate this observation in Fig. 12.3, where the spectra of two fragments from the **BTB** recordings versus the spectra of two fragments from the other boats are displayed.

Therefore, the conjecture is that the distribution of the energy (or some energy-like parameters) of acoustics signals from some class between different areas in the frequency domain, may provide a reliable characteristic signature for this class. In other words, they can be used as foundations for generating unique features that characterize the class of signals.

12.3 Formulation of the Approach

The discrete–time wavelet packets described in Sect. 9.1 provide a highly relevant tool for adaptive search of valuable frequency bands in a signal or a class of signals. Once implemented, the wavelet packet transform of a signal provides a collection of

different partitions of its Nyquist frequency band. The transform is computationally efficient. It was explained in Sect. 9.1.4 that energies in the blocks of the transform coefficients correspond to the energies in the respective subbands of the Nyquist frequency band of the signal. Thus, the set of energies in the blocks of the coefficients from the wavelet packet transform of a signal can be regarded as a "spectrum" of the frequency bands as opposed to the spectrum of single frequencies, which is provided by the Fourier transform.

By energy in a block of the wavelet packet coefficients, we understand the normalized l_2 norms of the coordinates vectors. To be specific, if projection of a signal $\mathbf{x} \in \Pi[N]$ onto the subspace $\Pi[N]_{m,l}$ is represented as

$$x_{m,l}[k] = \sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}[\lambda]\, \Psi_{m,l}^{2r}[k - 2^m\lambda]$$

then the respective energy is defined as

$$E_{m,l} \stackrel{\text{def}}{=} \frac{2^m}{N} \sqrt{\sum_{\lambda=0}^{N/2^m-1} \xi_{m,l}^2[\lambda]}.$$

The coefficients from the m-th decomposition level produce the frequency bands' "spectrum". The width of bands is, approximately, equal to $\nu/2^m$, where ν is the width of the Nyquist frequency band of the signal. We can see it, for example, in Fig. 12.4. This figure displays energies in the blocks of the wavelet packet coefficients from levels two to six of a boat acoustic signal. Here, the wavelet packet transform is based on orthogonal splines of eighth order.

Outline of the Approach

The pattern signals for the training phase and the online signals in the detection phase are formed by imposing a comparatively short window on each input signal followed by a shift of this window along the signal so that adjacent windows have some overlap. Then, the wavelet packet transform is applied to the segments and the energies in the selected coefficients blocks are calculated. As a result, each segment is represented by a set of energies.

To determine a few blocks of the wavelet packet transform coefficients, which discriminate a class of interest from other classes, we use a procedure similar to the Local Discriminant Bases method [11], which is a variation on the Best Basis algorithm [7, 12]. Energies in the selected blocks extracted from the segments of training signals are used as the reference sets for further application of the LDA classifier and for the design of the CART and *Aisles* classifiers.

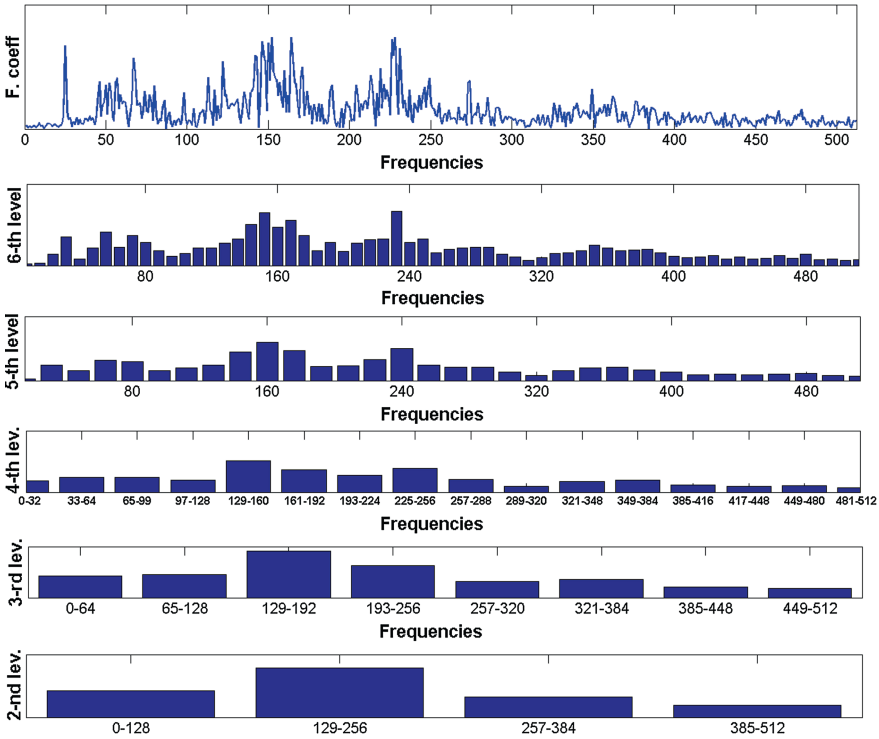


Fig. 12.4 *Top* Magnitude of the DFT of a **BTB** signal of length 1024 samples. *X*-axis: frequencies; *Y*-axis: magnitudes of the Fourier coefficients. Energies in the blocks of wavelet packet coefficients from the sixth decomposition level (second from the *top*, altogether 64 blocks) down to the second level (*bottom*, 4 blocks). Locations of the bars approximately correspond to the locations of the frequency subbands related to the blocks. *X*-axes: frequencies; *Y*-axes: values of energies in the blocks of wavelet packets coefficients

Segments, which are cut out from the test signals by the moving window, are identified individually in two steps:

1. The classifiers decide whether the segment is related to the *Boats* class or to the *Non-Boat* class: (**B** vs. **NB** decision).
2. If the decision is **B**, then the other set of classifiers decide whether the segment is related to the *BlueBoat* class or to the *OtherBoat* class: (**BTB** vs. **OB** decision).

Correspondingly, training the algorithm should be performed twice: once for **B** versus **NB** decision and once more for **BTB** versus **OB** decision. In the rest of the chapter, the **B** versus **NB** scheme is described. A couple of details about where the scheme **BTB** versus **OB** differs from it are indicated.

12.4 Description of the Algorithm and its Implementation

The classification (detection) algorithm consists of the following phases.

- Derivation of the classes signatures.
- Training the classifiers.
- Identification of test data.

12.4.1 Derivation of the Classes Signatures

In the given setting, the signatures are derived in the following way:

12.4.1.1 Construction of the Training Databases

- Part of the available data is selected for the training (**Tr**-data) phase, the rest is left for the test (**Te**-data) phase.
- Using video recordings for reference, pieces related to **B** and **NB** classes are clipped from **Tr**-data;
- The selected pieces are sliced into overlapping segments of length l (typically, $l = 1024$);
- The training database **B–NB** is prepared for **B** versus **NB** discrimination. The **B–NB** database consists of two matrices: $\mathbf{M}_b$,—of size $m_b \times l$, where the rows are the **B**-segments and $\mathbf{M}_n$,— of size $m_n \times l$, where the rows are the **NB**-segments.

12.4.1.2 Transformation of the Training Data

- The wavelet packet transform down to level J (typically, $J = 6$) is applied to rows of the matrices $\mathbf{M}_b$ and $\mathbf{M}_n$.
- The energies in the blocks of the transform coefficients are calculated. Altogether, there are $K = 2^{J+1} - 1$ blocks of coefficients. Thus, each segment is represented by an energy vector of length K. This operation reduces the dimensionality of the segments from l to K (typically, from 1024 to 127). Subsequent operations further reduce the dimensionality;
- The energy vectors are arranged into the matrices $\mathbf{W}_b$, of size $m_b \times K$ and $\mathbf{W}_n$, of size $m_n \times K$. A row represents a segment.

12.4.1.3 Determination of the Signature(s) of the Class(es) of Interest

- The averages $\overline{\mathbf{W}}_b$ and $\overline{\mathbf{W}}_n$ along the columns of the energy matrices $\mathbf{W}_b$ and $\mathbf{W}_n$ are calculated, respectively. We call the vectors $\overline{\mathbf{W}}_b$ and $\overline{\mathbf{W}}_n$ the energy maps

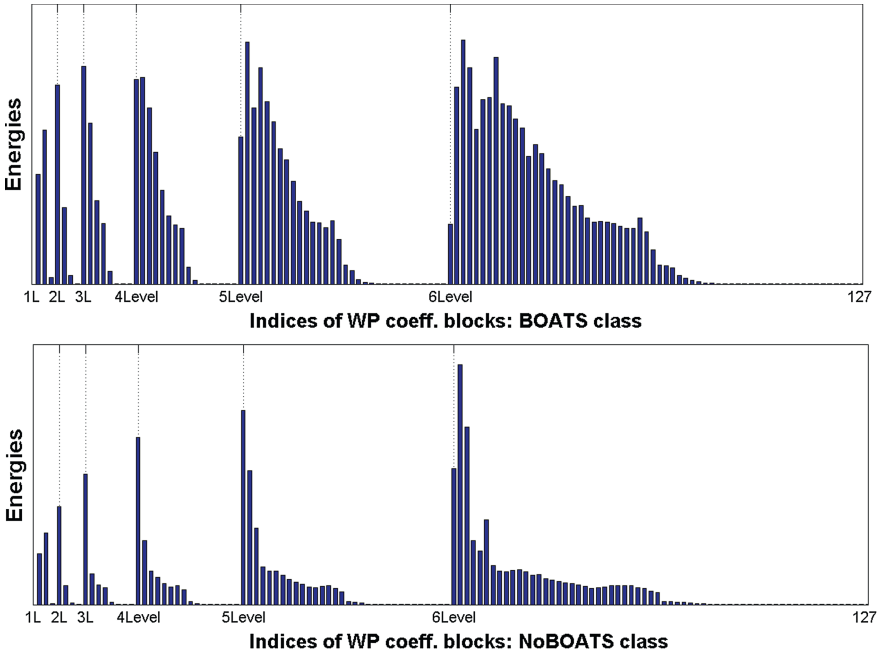


Fig. 12.5 Energy maps of the classes **B** and **NB**. *X*-axes: indices of blocks of wavelet packet coefficients; *Y*-axes: averaged values of energy in the blocks. *1* original signal, *2*, *3* indices of the 1-st level blocks; *4*,...,*7* 2-nd level; *8*,...,*15* 3-rd level; *16*,...,*31* 4-th level; *32*,...,*63* 5-th level; *64*,...,*127* 6-th level. *Top* map for the *Boats* class; *Bottom* map for the *NoBoatss* class

of the classes **B** and **NB**, respectively. Typical energy maps are displayed in Fig. 12.5.

- The differences $\mathbf{D}_{bn} \stackrel{\text{def}}{=} \overline{\mathbf{W}}_b - \overline{\mathbf{W}}_n$ are calculated. The difference of the above energy maps is displayed in Fig. 12.6.
- The discriminating power of a certain block of coefficients is defined as the magnitude of the difference $\mathbf{D}_{bn}$ of this block;
- There is *parent–offsprings* dependence between the blocks of wavelet packets coefficients (Table 9.6). Two offsprings are related to the same frequency bands as their parent. It is clearly seen in Fig. 12.4. For example, the pair of blocks #3 and #4 from the third decomposition level are related to the same frequency band as their parent: the block #2 from the second level.
- Comparing parent–offsprings discriminating power. If the parent is "stronger" then the offsprings are discarded and vice versa;
- When frequency overlap is removed, a small number M_{bn}, whose discriminating power is the strongest (typically, $M_{bn} = 6 \div 10$) is selected out of the remaining blocks. Denote by L_{bn} the list of their indices. The list is considered as the signature of the class **B** against **NB**. Energies in the selected blocks of coefficients serve as characteristic features of the segments.

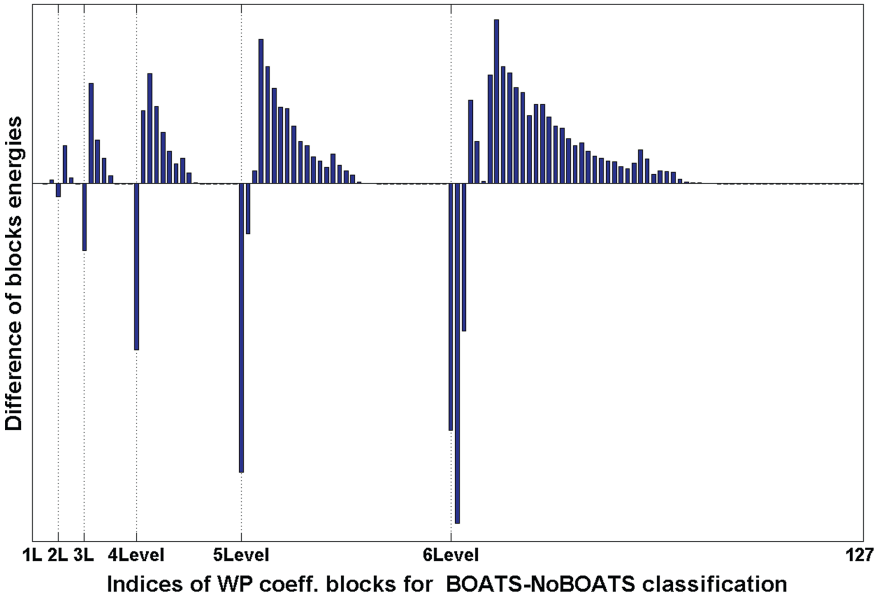


Fig. 12.6 Difference of the energy maps between the classes **B** and **NB**. *X*-axis: indices of blocks of wavelet packet coefficients; *Y*-axis: differences of averaged values of energy in the blocks. *1* original signal, *2*,*3* indices of the 1-st level blocks; *4*,...,*7* 2-nd level; *8*,...,*15* 3-rd level; *16*,...,*31* 4-th level; *32*,...,*63* 5-th level; *64*,...,*127* 6-th level

- The energy matrices $\mathbf{W}_b$ and $\mathbf{W}_n$ are reduced to features matrices $\widehat{\mathbf{W}}_b$ of size $m_b \times M_{bn}$ and $\widehat{\mathbf{W}}_n$ of size $m_n \times M_{bn}$ by retaining only the columns listed in L_{bn}.
- As a result, each segment of length l is represented by the M_{bn}-tap vector of energies in the selected blocks of the wavelet packet transform coefficients. Thus, its dimensionality becomes reduced from l to M_{bn} (typically, from 1024 to 8).

12.4.1.4 Refinement of the Feature Matrices

In order to refine the feature matrices $\widehat{\mathbf{W}}_b$ and $\widehat{\mathbf{W}}_n$, their rows are tested. Recall that each row in $\widehat{\mathbf{W}}_b$ is associated with a segment in a **B**-class signal and each row in $\widehat{\mathbf{W}}_n$ is associated with a segment in an **NB**-class signal. We calculate the Mahalanobis distances [9, 10] d^{bb} and d^{bn} of each row in the matrix $\widehat{\mathbf{W}}_b$ from the matrices $\widehat{\mathbf{W}}_b$ and $\widehat{\mathbf{W}}_n$, respectively. It may happen that, for some rows from $\widehat{\mathbf{W}}_b$ the distance d^{bb} to the "native" matrix exceeds the distance d^{bn} to the matrix $\widehat{\mathbf{W}}_n$. Such rows are removed from the matrix $\widehat{\mathbf{W}}_b$. The same is done for the features matrix $\widehat{\mathbf{W}}_n$. The refined features matrices are denoted as $\mathbf{R}_b$ and $\mathbf{R}_n$, respectively. They are called the *reference matrices* for the **B** versus **NB** classification.

Typical reference matrices $\mathbf{R}_b$ and $\mathbf{R}_n$ are displayed in Fig. 12.7. Even visually, energies in the eight selected blocks for the **B** class segments differ from those for the **NB** class segments.

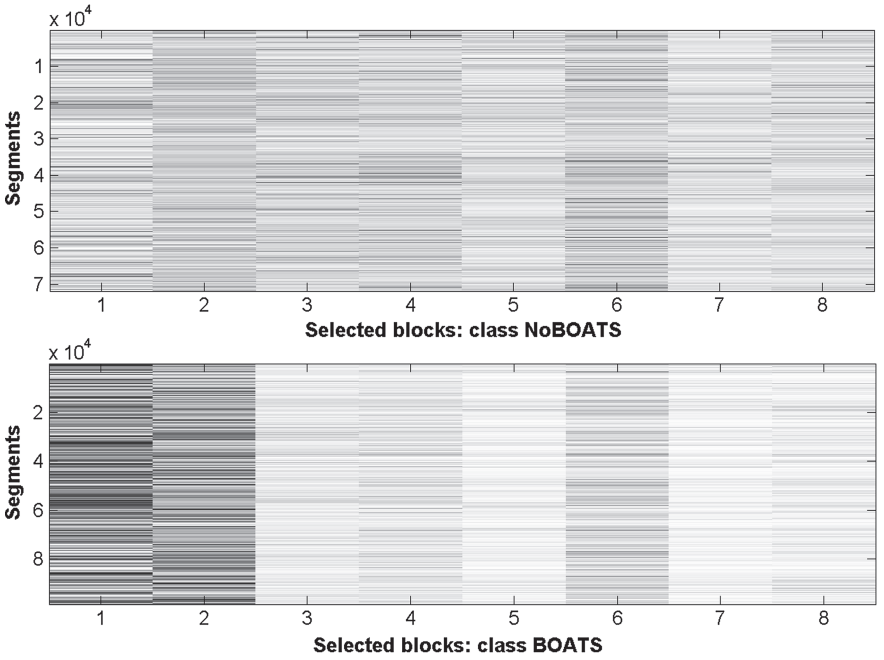


Fig. 12.7 Reference matrices $\mathbf{R}_b$ and $\mathbf{R}_n$ for **B** versus **NB** classification, which represent energies in 8 selected blocks of the wavelet packet coefficients for segments from *NoBoats* recordings (*top*) and *Boats* recordings (*bottom*). Rows represent slices, columns represent blocks

12.4.1.5 BTB Versus OB Classification

Exactly the same procedures as above are applied in order to produce the list S_{bo} of M_{bo} indices, which is the signature for the class **BTB** against **OB**. Similarly, the reference matrices $\mathbf{R}_{btb}$ and $\mathbf{R}_{ob}$ are produced. Naturally, the dissimilarity between **BTB** signals and **OB** signals is much less than the dissimilarity between **B** signals and **NB** signals. Thus, the energy maps $\overline{\mathbf{W}}_{btb}$ and $\overline{\mathbf{W}}_{ob}$, which are displayed in Fig. 12.8, are hardly distinguishable. Therefore, the *refinement* procedure for the **BTB** versus **OB** classification is stricter than the refinement for **B** versus **NB**. Thus updated, the method, which derives the class signature, proved to be successful in this case.

12.4.1.6 Summary

As a result of the above operations, the dimensionality of the training set is substantially reduced. Typically, a segment of length 1024 is embedded into an 8-component features vector. Ostensibly, this part of the process looks computationally expensive, especially if, for better robustness, large training sets are involved. This procedure is called once and it is done offline before the detection and classification phase that is done online (real time).

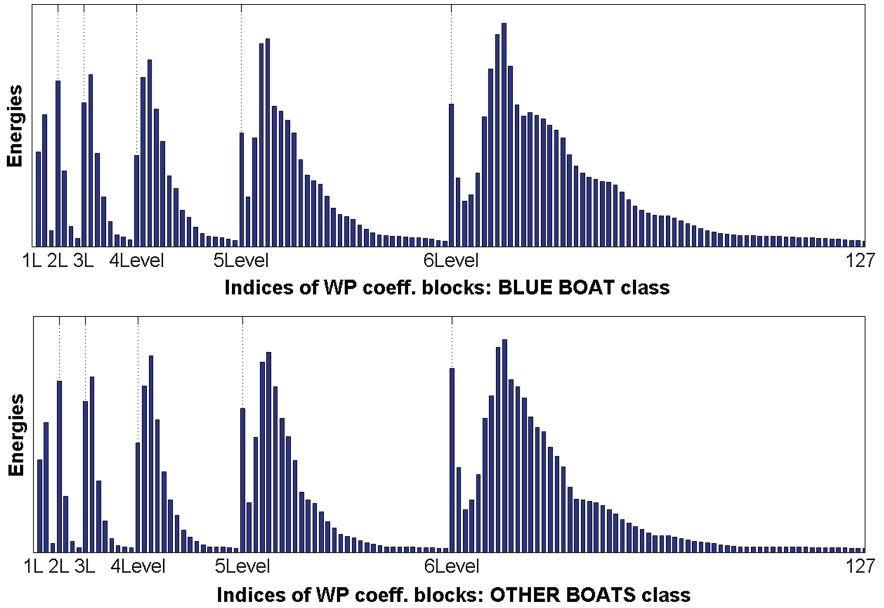


Fig. 12.8 Energy maps of classes **BTB** and **OB**. *X*-axes: indices of blocks of wavelet packet coefficients; *Y*-axes: averaged values of energy in the blocks. *1* original signal, *2*,*3* indices of the 1-st level blocks; *4*,...,*7* 2-nd level; *8*,...,*15* 3-rd level; *16*,...,*31* 4-th level; *32*,...,*63* 5-th level; *64*,...,*127* 6-th level. *Top* map for the *Blue Tourist Boat* class; *Bottom* map for the *Other Boats* class

The classes signatures S_{bn} and S_{bo} and the pairs $\mathbf{R}_b$ – $\mathbf{R}_n$ and $\mathbf{R}_{btb}$ – $\mathbf{R}_{ob}$ of reference matrices are stored for further use.

12.4.2 Training the Classifiers

Once two pairs of the reference matrices $\mathbf{R}_b - \mathbf{R}_n$ and $\mathbf{R}_{btb} - \mathbf{R}_{ob}$ have been collected, the classifiers should be trained. For this purpose, features vectors, which form rows in the reference matrices are used.

12.4.2.1 Classification and Regression Trees

First, the features vectors are used as the input patterns for the construction of the CARTs.

The construction of the tree is done by a binary split of the space of input patterns $X \longrightarrow \{X_1 \bigcup X_2 \bigcup \ldots \bigcup X_r\}$, where $\{X_k\}$ are the terminal nodes of the tree. Once a vector appears in the node X_k, its membership can be predicted with a sufficient reliability. The answer is the class the vector is assigned to and the probability of

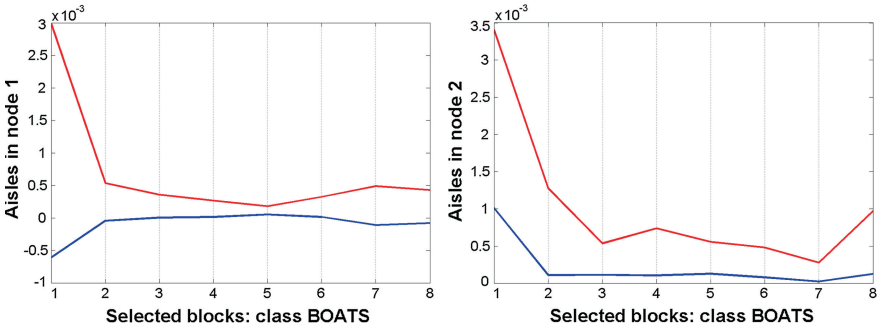


Fig. 12.9 Examples of *Aisles* in two different CART nodes for **B**–**NB** classification. X-axes: indices $k = 1, ..., 8$ of selected blocks; *upper diagrams*: $\bar{m}^k + 2\bar{s}^k$, *lower diagrams*: $\bar{m}^k - 2\bar{s}^k$

this assignment. The basic idea behind this split is that the data in each descendant subset is "purer" than the data in the parent subset. The scheme is described in full details in [6]. A brief outline is given in the appendix to this chapter.

As a result, two trees $\mathbf{T}_{bn}$ and $\mathbf{T}_{bo}$ are designed for the discrimination **B** versus **NB** and **BTB** versus **OB**, respectively.

12.4.2.2 Refinement of CART: *Aisles*

Once a feature vector is presented to the CART tree $\mathbf{T}_{bn}$, it is dispatched to a terminal node. Each terminal node is assigned to one of the classes **B** or **NB**. To reduce false alarms, the CART classifier is refined in the following way.

All the rows of the *"Boat"* reference matrix $\mathbf{R}_b$ are presented to the tree $\mathbf{T}_{bn}$. Each row is dispatched to one of the terminal nodes that are associated with the **B** class. Assume that the set $\mathbf{R}_b^k$ of rows from the matrix $\mathbf{R}_b$ was dispatched to a node T_b^k. We calculate the M_{bn}-tap vectors $\mathbf{m}^k = \left\{m_1^k, ..., m_{M_{bn}}^k\right\}$, which is the mean value of the rows $\mathbf{R}_b^k$, and $\mathbf{s}^k = \left\{s_1^k, ..., s_{M_{bn}}^k\right\}$, which is the standard deviation (STD) of these rows. Then, we form the aisle $\mathbf{A}_b^k = \bar{m}^k \pm 2\bar{s}^k$. This is done for all the **B**-terminal nodes T_b^k, $k = 1, ..., K_b$.

Similarly, we present all the rows in the reference matrix $\mathbf{R}_{btb}$ to the tree $\mathbf{T}_{bo}$ in order to form the aisles A_{btb}^k for all the **BTB**-terminal nodes T_{btb}^k, $k = 1, ..., K_{btb}$, in the tree $\mathbf{T}_{bo}$.

Two examples of *Aisles* are displayed in Fig. 12.9.

12.4.2.3 Summary

As a result from the training operations, the two lists, L_{bn} and L_{bo}, of the characteristic blocks are compiled for the discrimination **B** versus **NB** and **BTB** versus **OB**,

respectively, the pairs $\mathbf{R}_b - \mathbf{R}_n$ and $\mathbf{R}_{btb} - \mathbf{R}_{ob}$ of the reference matrices are collected, the trees $\mathbf{T}_{bn}$ and $\mathbf{T}_{bo}$ and two sets $\mathbf{A}_b \stackrel{\text{def}}{=} \left\{A_b^k\right\}_{k=1}^{K_b}$ and $\mathbf{A}_{btb} \stackrel{\text{def}}{=} \left\{A_{btb}^k\right\}_{k=1}^{K_{btb}}$ of aisles are designed. All these data are stored to be used to identify newly arrived signals.

12.4.3 Identification of an Acoustic Signal

Identification of a newly arrived acoustic recording, which did not participate in the training phase, is implemented in three steps:

1. Preprocessing

The process is exactly the same as in the preprocessing of the training signals. Once a signal is assigned for identification, the following steps are carried out:

- It is sliced into overlapping segments of length l;
- The wavelet packet transform is applied down to scale J to each segment;
- Energies in the blocks of the transform coefficients are calculated;
- The segments are embedded into energy vectors of length $K = 2^{J+1} - 1$.

2. Primary identification

The *Boats* (**B**) segments are distinguished from the *NoBoats* (**NB**) segments in the following way:

- The energy vectors are reduced to features vectors by selecting M_{bn} blocks whose indices are in the signature list L_{bn}.
- Each features vector is classified by the Linear Discriminant Analysis (LDA) using the reference matrices $\mathbf{R}_b$ and $\mathbf{R}_n$.
- Each features vector is classified by CART using the $\mathbf{T}_{bn}$ tree.
- Once a features vector is assigned to the **B** class by CART, it is subjected to the test by *Aisles* using the $\mathbf{A}_b$ set. Assume the vector $\mathbf{v} = \left\{v_1, ..., v_{M_{bn}}\right\}$ is dispatched to the node T_b^k. If the vector "passes through" the aisle A_b^k, that is $m_l^k - 2s_l^k \le v_l \le m_l^k + 2s_l^k,\ l = 1, ..., M_{bn}$, then the *Aisles* classifier assigns this vector (segment) to the **B** class. Otherwise, it is assigned to the **NB** class. This additional check reduces the false alarm rate.
- The answer from the **B** versus **NB** classifiers is 1 if the vector (segment) is classified as **B** and 0 otherwise;
- If at least one out of three answers for a given segment is zero then the segment is classified as **NB**: otherwise it is classified as **B**.

3. Final identification of B segments

This means separation of the **BTB** segments from the **OB** segments.

Once a segment is classified as *Boats*:

- The corresponding energy vector is reduced to the features vector by selecting M_{bo} blocks whose indices are in the signature list L_{bo}.
- The features vector is classified by LDA using the reference matrices $\mathbf{R}_{btb}$ and $\mathbf{R}_{ob}$.
- The features vector is classified by CART using the $\mathbf{T}_{bo}$ tree.
- If the features vector is assigned to the **BTB** class by CART then it is tested by *Aisles* using the $\mathbf{A}_{btb}$ set. If the vector "passes through" the corresponding aisle then the *Aisles* classifier assigns this vector (segment) to the **BTB** class. Otherwise, it is assigned to the **OB** class.
- The answer from the classifiers is 1 if the vector (segment) is classified as **BTB** and 1/2 otherwise.
- For a segment s_k, two decisions units are compiled: $\mathbf{DEC}_k^1$ is the product of LDA and CART answers for a given segment, whereas $\mathbf{DEC}_k^2$ is the product of LDA and *Aisles* answers. These decisions assume the following values

$$\mathbf{DEC}_k^i = \begin{cases} 1, & \text{if both classifiers give positive answers;} \\ 1/2, & \text{if classifiers give different answers;} \\ 1/4, & \text{if both classifiers give negative answers} \\ 0, & \text{if the segment was primarily classified as } \mathbf{NB}. \end{cases}, \quad i = 1, 2. \tag{12.1}$$

12.5 Examples

A series of experiments was conducted to detect the presence of the *Blue Tourist Boat* from acoustic signals recorded by a single hydrophone in an harbor. Half of the available recordings were used for training the algorithm and the rest was left for testing. The background was highly variable: there were fragments with no boats, and the water conditions and surrounding noise varied. Fragments of recordings with many different types of boats, which were cruising in the area, also constituted the background. The stationary hydrophone recorded sounds emitted by the **BTB** and other boats from different locations and directions, from different distances from the receiver and from different modes of the engines' operation. No special actions were carried out to improve the quality of the recording, which was far from perfect.

In spite of these variabilities and drawbacks, the presented algorithm demonstrated robust detection of the target with minimal false alarm. The following three figures visualize the performance of the algorithm.

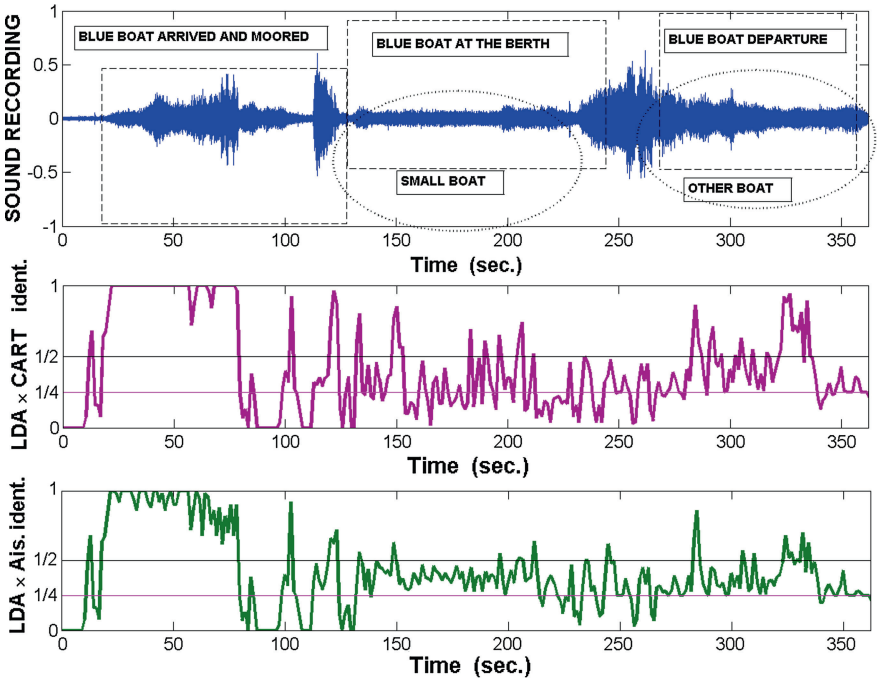


Fig. 12.10 Identification of recording # 1300. X-axes—time (seconds) from the start of recording. *Top frame* recording of the sound. *Central frame* averaged product of answers from LDA and CART classifiers on identification of short segments of the signal. *Bottom frame* averaged product of answers from LDA and *Aisles* classifiers on identification of short segments of the signal. While the curves are located above 1/2, the corresponding segments are classified as **BTB**; localization between 1/2 and 0 means **OB**; and the intervals where curves touche zero are classified as **NB**

12.5.1 Structure of Presented Figures

Figures 12.10, 12.11 and 12.12 display the identification results of three acoustic recordings, which did not participate in the training procedures. Each figure consists of three frames. In all the frames, the X-axes indicate the time (seconds) from the start of the recording. The top frame in each figure displays the recorded signal. Rectangles mark the time intervals where the **BTB** is present. Ellipses mark the time intervals where **OB** are present.

The curves in the central and bottom frames present the detection results in the following way:

Central frame: The array $\{\mathbf{DEC}_k^1\}$ of the decisions was smoothed by a forward and backward moving average followed by decimation. The remaining values were linked by a curve. The curve was synchronized with the input acoustic signal.

Bottom frame: The same was done with the array $\{\mathbf{DEC}_k^2\}$.

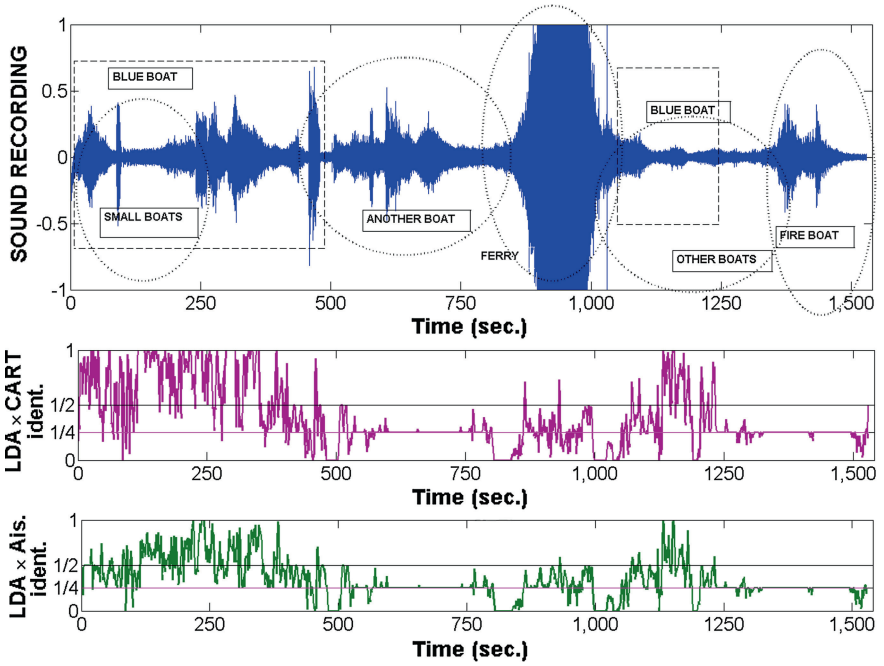


Fig. 12.11 Identification of recording # 1129. *X*-axes—time (seconds) from the start of the recording. *Top frame* recording of the sound. *Central frame* averaged product of answers from LDA and CART classifiers on identification of short segments of the signal. *Bottom frame* averaged product of answers from LDA and *Aisles* classifiers on identification of short segments of the signal. While the curves are located above 1/2, the corresponding segments are classified as **BTB**; localization between 1/2 and 0 means **OB**; and the intervals where curves touche zero are classified as **NB**

The horizontal lines $L_{1/2}$ and $L_{1/4}$ at heights 1/2 and 1/4, respectively, are drawn for reference only. If, at some interval, the curve lies above the line $L_{1/2}$, it means that both the decision units have detected the target at this interval. If the curve lies between the lines $L_{1/2}$ and $L_{1/4}$ then the units have made different decisions. If the curve's values are positive but do not exceed 1/4 then it means that, at the corresponding interval, both classifiers have detected an **OB** other than the **BTB**.

12.5.2 Display of Results

Recording # 1300 lasted for about 6 min. Most of the time the **BTB** was present at the scene but its state varied. Both $\mathbf{DEC}^1$ and $\mathbf{DEC}^2$ detected its arrival and its approach to the berth. While it was standing at the berth (with the working engine), a small boat passed by the receiver at a high speed. The decision unit $\mathbf{DEC}^1$ identified many segments of this mixed interval as belonging to **BTB**, however the unit $\mathbf{DEC}^2$, which produced the answer from *Aisles*, was much less tolerant. When the **BTB** was

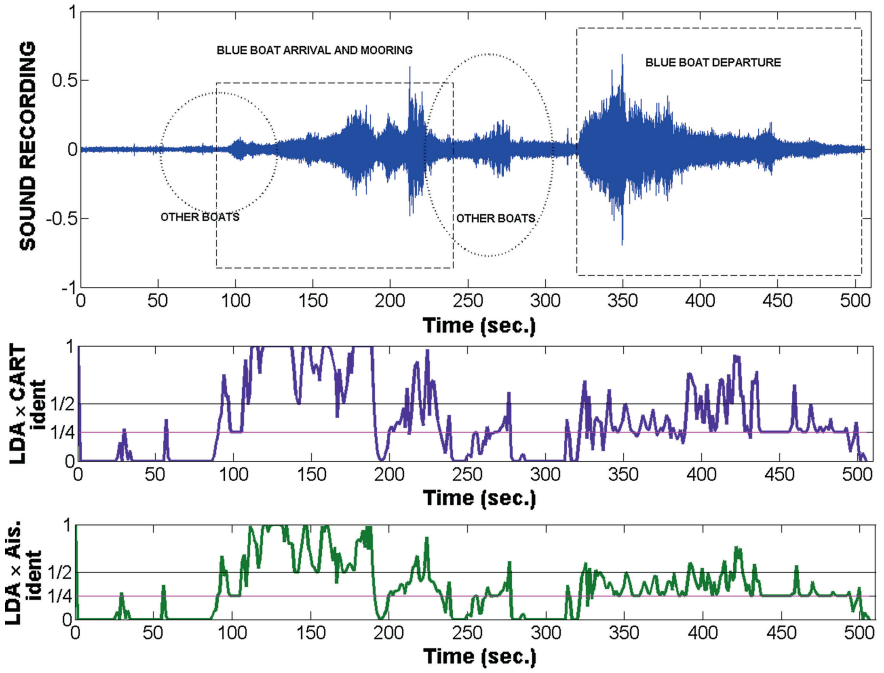


Fig. 12.12 Identification of recording # 1555. *X*-axes—time (seconds) from the start of the recording. *Top frame* recording of the sound. *Central frame* averaged product of answers from LDA and CART classifiers on identification of short segments of the signal. *Bottom frame* averaged product of answers from LDA and *Aisles* classifiers on identification of short segments of the signal. While the curves are located above 1/2, the corresponding segments are classified as **BTB**; localization between 1/2 and 0 means **OB**; and the intervals where curves touche zero are classified as **NB**. The sea is choppy

departing, another boat was present in the area. At some times, the sound from the **BTB** prevailed, at other times the sound from the other boat was stronger. Therefore, part of the segments at this interval were identified as **BTB**, while the rest were declared as being **OB**. Clearly, as to the purity of the signal, $\mathbf{DEC}^2$ was much more exigent than $\mathbf{DEC}^1$.

Recording # 1129 continued for about 25 min. At the outset, the **BTB** arrived and moored. After a while it departed. On arrival, it was accompanied by small boats. On departure, another boat arrived. Then, after a short **NB** interval, a huge ferry boat sailed into the area. When the ferry boat disappeared, a small speed boat arrived followed by the **BTB**, which passed by the receiver at a long distance. At the same time, another boat was present at the scene. In the end, the receiver recorded a strong signal emitted by a fire boat. All these events were properly handled by both decision units. $\mathbf{DEC}^2$ produced fewer false alarm than $\mathbf{DEC}^1$ but missed a few events when the **BTB** was present.

Recording # 1555 lasted for about eight minutes while sea was choppy. There were no boats till about 90 s from the start of the recording. The decision units marked almost all the segments from this interval as **NB**. Then, a couple of small motor boats arrived at the scene followed by the **BTB**, which approached the berth. While it was standing at the berth, another boat was passing by. $\mathbf{DEC}^1$ mistakenly attributed a few segments from the latter event to **BTB**, but $\mathbf{DEC}^2$ corrected this. After a calm interval, which was classified as **NB**, the **BTB** departed. During its departure, the **BTB** was detected by $\mathbf{DEC}^1$, but the reports from $\mathbf{DEC}^2$ were ambivalent.

Comments:

- The detection experiments demonstrated the relevance of the approach based on wavelet packets to feature extraction.
- Combination of the LDA with CART classifiers ($\mathbf{DEC}^1$) is more sensitive to the presence of the target in comparison to the LDA–*Aisles* combination ($\mathbf{DEC}^2$). Nevertheless, it sometimes produces false alarms. On the other hand, the decision unit $\mathbf{DEC}^2$, while significantly reducing false alarms, sometimes missed the target.
- The algorithm produced satisfactory detection and classification results even when the conditions of the real signals were essentially different from those of the training data. When the conditions of the captured signals were close to the training conditions, the detection was almost perfect.
- Typically, arrival of the target boat was detected much better in comparison to its detection upon departure.

Appendix: Outline of the CART Algorithm

A comprehensive exposition of the CART scheme can be found in [6]. We consider a two-class classification problem.

Building the tree

The space X of input patterns from the reference set consists of two reference matrices $V^l,\ l = 1, 2$ of sizes $\mu_l \times n$, respectively. For simplicity, assume that $\mu_1 = \mu_2$. The i-th row of the matrix V^l is a vector $V^l(i, :)$ of length n representing the signal s_i^l ,which belongs to the class C^l. In our case, n is equal to the number of discriminant blocks. All row vectors $V^l(i, :)$ should be normalized, i.e.

$$0 \le V^l(i, j) \le 1, \qquad \sum_{j=1}^{t} V^l(i, j) = 1.$$

The tree-structured classifier to be constructed has to divide our space X into J disjoint subspaces

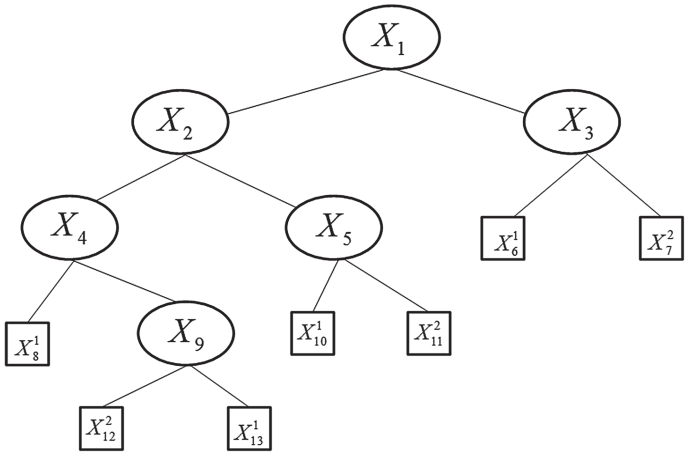


Fig. 12.13 Example of a tree for two-class problem. The terminal nodes are indicated by *rectangular boxes* and are designated by a class label; non-terminal nodes are indicated by *circles*

$$X = \bigcup_{\nu=1}^{J} X_\nu^t. \tag{12.2}$$

Each subspace X_ν^t must be "pure" in the sense that the percentage of vectors from one of the matrices V^l, must prevail over the percentage of the vectors from the other matrix. (In the original space X both are 50 %.)

The construction of the binary tree is started by a split of X into two descendant subspaces:

$$X = X_1 = X_2 \bigcup X_3, \quad X_2 \bigcap X_3 = \emptyset.$$

To do so, CART chooses a split variable y_j and split value z_j to achieve minimal possible "impurity" of the subspaces X_2 and X_3. The split rule for the space X_1 is:

If a vector $y = (y_1, ..., y_n)$ *satisfies the condition* $y_j \leq z_j$ *, then it is transferred to the subspace* X_2 *, otherwise it is transferred to the subspace* X_3*. In addition, we divide the subspace* X_2 *in a similar manner*:

$$X_2 = X_4 \bigcup X_5, \quad X4 \bigcap X_5 = \emptyset.$$

The subsequent split variable y_k and split value z_k are selected on the basis that the data in each of the descendant subspaces were "purer" than the data in the parent subspace. Then one of the subspaces X_4 or X_5 can be further divided recursively until we reach the so-called terminal subspace X_1^t which is not split further. The decisions about whether a subspace is classified as terminal subspace depends on the predetermined minimal "impurity" and the minimal size of the subspace. The terminal subspace X_1^t is assigned to the class C^l, with the probability

$$p_1^l = \frac{m_1^l}{m_1} = \frac{\#\{y \in X_1^t \bigcap V^l\}}{\#\{y \in X_1^t\}},$$

where m_1^l is the number of points in node X_1^t that belongs to class C^l and m_i is the total number of points in the subspace X_1^t. After termination is reached in the subspace X_1^t we return to subspace X_3 which was not split so far. Similarly, we reach the next terminal subspace X_2^t. We do the same with one of yet non-split subspaces and finally the tree of (12.2) is constructed. In the terminology of graph theory, the space X is called the root node, and the non-terminal and terminal subspaces are the nonterminal and terminal nodes. This process is illustrated in Fig. 12.13. The terminal nodes are marked as rectangles.

Classification

A vector $x = (x_1, ...x_n)$, which is fed to the tree, in the first step is assigned to either node X_2 in the case where the coordinate $x_j \leq z_j$ or to node X_3 otherwise. Finally, by checking subsequent split variables, the vector is forwarded into a terminal node X_r^t which is labeled as class C^l, with probability p_r^l.

References

1. A. Averbuch, K. Hochman, N. Rabin, A. Schclar, V. Zheludev, A diffusion framework for detection of moving vehicles. Digit. Signal Process. **20**(1), 111–122 (2010)
2. A. Averbuch, E. Hulata, V. Zheludev, I. Kozlov, A wavelet packet algorithm for classification and detection of moving vehicles. Multidimension. Syst. Signal Process. **12**(1), 9–31 (2001)
3. A. Averbuch, I. Kozlov, V. Zheludev, in Wavelets: Applications in Signal and Image Processing IX, eds. by A.F. Laine, M.A. Unser, A. Aldroubi. Wavelet-packet-based algorithm for identification of quasi-periodic signals, Proceedings of SPIE, vol. 4478, 2001, pp. 353–360.
4. A. Averbuch, V. Zheludev, P. Neittaanmäki, P. Wartiainen, K. Huoman, K. Janson, Acoustic detection and classification of river boats. Appl. Acoust. **72**(1), 22–34 (2011)
5. A. Averbuch, V. Zheludev, N. Rabin, A. Schclar, Wavelet-based acoustic detection of moving vehicles. Multidimension. Syst. Signal Process. **20**(1), 55–80 (2009)
6. L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, *Classification and Regression Trees* (Chapman & Hall, New York, 1993)
7. R.R. Coifman, V.M. Wickerhauser, Entropy-based algorithms for best basis selection. IEEE Trans. Inform. Theor. **38**(2), 713–718 (1992)
8. R.A. Fisher, The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936)
9. P.C. Mahalanobis, On the generalised distance in statistics. Proc. Nat. Inst. Sci. India **2**(1), 49–55 (1936)
10. G.J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition* (Wiley Interscience, Philadelphia, 1992)
11. N. Saito, R.R. Coifman, Local discriminant bases and their applications. J. Math. Imaging Vision **5**(4), 337–358 (1995)
12. M.V. Wickerhauser, *Adapted Wavelet Analysis: From Theory to Software* (AK Peters, Wellesley, 1994)

Chapter 13
Periodic Discrete Splines

Abstract Periodic discrete splines with different periods and spans were introduced in Sect. 3.4. In this chapter, we discuss families of periodic discrete splines, whose periods and spans are powers of 2. As in the polynomial splines case, the Zak transform is extensively employed. It results in the Discrete Spline Harmonic Analysis (DSHA). Utilization of the Fast Fourier transform (FFT) enables us to implement all the computations in a fast explicit way.

13.1 One-dimensional Discrete Spline Harmonic Analysis

We will operate on centered discrete splines of even order $p = 2r$, which are $N = 2^j$-periodic and whose span is a power of 2: $K = 2^m$. The powers j and m are natural numbers and $j \gg m$. The space of N-periodicsignals is denoted by $\Pi[N]$. As before, $\omega \stackrel{\text{def}}{=} e^{2\pi\, i/N}$ and $N_m \stackrel{\text{def}}{=} N/2^m$.

13.1.1 Discrete Periodic Splines Spaces

The centered periodic B-spline of order $p = 2r$ of span $K = 2^m$ denoted by $\mathbf{b}^{2r}_{[m]} = \left\{b^{2r}_{[m]}[k]\right\}$, $k \in \mathbb{Z}$, is derived by N-periodization of the discrete B-spline $\check{\mathbf{b}}^{2r}_{[m]}$ supported on the set $-r\,2^m + 1, \ldots, r\,2^m - 1$, which was defined in Sect. 3.4.1:

$$b^{2r}_{[m]}[k] \stackrel{\text{def}}{=} \sum_{l\in\mathbb{Z}} \check{b}^{2r}_{[m]}[k + Nl].$$

Its DFT

$$\hat{b}^{2r}_{[m]}[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, b^{2r}_{[m]}[k] = \left(\frac{\sin(2^m\pi\, n/N)}{2^m\, \sin(\pi\, n/N)}\right)^{2r}, \quad \hat{b}^{2r}_{[m]}[0] = 1,$$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_13,

is a cosine polynomial. Thus,

$$b^{2r}_{[m]}[k] = \frac{1}{N}\sum_{n=0}^{N-1} \omega^{kn}\, \hat{b}^{2r}_{[m]}[n] = \frac{1}{N}\sum_{n=0}^{N-1} \omega^{kn} \left(\frac{\sin(2^m \pi\, n/N)}{2^m\, \sin(\pi\, n/N)}\right)^{2r}. \tag{13.1}$$

Surely, $\hat{b}^{0}_{[m]}[n] = 1$ for all n.

Proposition 13.1 *There holds a recursive relation between the DFT of B-splines of different spans*

$$\hat{b}^{2r}_{[m+1]}[n] = \hat{b}^{2r}_{[m]}[n]\, \cos^{2r} \frac{2^m \pi\, n}{N}, \quad m = 0, 1, \ldots, \tag{13.2}$$

Proof The DFT of the spline $\mathbf{b}^{2r}_{[m+1]}$ is

$$\begin{aligned}
\hat{b}^{2r}_{[m+1]}[n] &= \left(\frac{\sin(2^{m+1} \pi\, n/N)}{2^{m+1}\, \sin(\pi\, n/N)}\right)^{2r} \\
&= \left(\frac{2\cos(2^m \pi\, n/N)\, \sin(2^m \pi\, n/N)}{2^{m+1}\, \sin(\pi\, n/N)}\right)^{2r} \\
&= \hat{b}^{2r}_{[m]}[n]\, \cos^{2r} \frac{2^m \pi\, n}{N}.
\end{aligned}$$

■

Denote by ${}^{2r}\mathscr{S}_{[m]}$ the space of N-periodic discrete splines $\mathbf{s}_{[m]} = \left\{s_{[m]}[k]\right\}$, $k \in \mathbb{Z}$, which are represented via circular shifts of the periodic B-splines

$$s_{[m]}[k] = \sum_{l=0}^{N_m-1} q[l]\, b^{2r}_{[m]}[k - 2^m l], \tag{13.3}$$

where $\mathbf{q} = \{q[k]\}$, $k \in \mathbb{Z}$, is an N_m-periodic sequence. The DFT of the discrete spline $\mathbf{s}^{2r}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$

$$\begin{aligned}
\hat{s}^{2r}_{[m]}[n] &= \sum_{k=0}^{N-1} \omega^{-nk} \sum_{l=0}^{N_m-1} q[l]\, b^{2r}_{[m]}[k - 2^m l] = \sum_{l=0}^{N_m-1} \omega^{2^m nl}\, q[l] \sum_{k=0}^{N-1} \omega^{-nk}\, b^{2r}_{[m]}[k] \\
&= \hat{q}[n]_m \left(\frac{\sin(2^m \pi\, n/N)}{2^m\, \sin(\pi\, n/N)}\right)^{2r}, \quad \hat{q}[n]_m = \sum_{l=0}^{N_m-1} \omega^{2^m nl}\, q[l].
\end{aligned} \tag{13.4}$$

Remark 13.1.1 The space $\Pi[N]$ of N-periodic signals can be regarded as the space $\mathcal{S}_{[0]}$ of splines $\mathbf{s}_{[0]}$ of arbitrary order whose span is zero, where $b_{[0]}[k] \stackrel{\text{def}}{=} \delta[k](\text{mod}(N)$. Consequently, each signal $\mathbf{x} \in= \mathcal{S}_{[0]}$ is represented as

$$\mathbf{x} = \sum_{l=0}^{N-1} x[l]\delta[k-l] \Longleftrightarrow \mathbf{x} = \sum_{l=0}^{N-1} x[l]\mathbf{b}_{[0]}[k-2^0 l].$$

Equation (13.2) remains true for $m = 0$.

13.1.2 Exponential Discrete Periodic Splines

The Zak transforms of the periodic discrete B-splines, which are called discrete periodic exponential splines, are defined in Eq. (3.40). In our case, $K = 2^m$.

$$\zeta^{2r}_{[m]}(\nu) = \left\{\zeta^{2r}_{[m]}(\nu)[k]\right\}, \quad \zeta^{2r}_{[m]}(\nu)[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N_m-1} \omega^{2^m \nu l}\, b^{2r}_{[m]}[k-2^m l]. \quad (13.5)$$

Substituting Eq. (13.1), we have

$$\begin{aligned}
\zeta^{2r}_{[m]}(\nu)[k] &= \frac{1}{N}\sum_{l=0}^{N_m-1} \omega^{2^m \nu l} \sum_{n=0}^{N-1} \omega^{(k-2^m l)n}\, \hat{b}^{2r}_{[m]}[n] \\
&= \frac{1}{N}\sum_{n=0}^{N-1} \omega^{kn}\, \hat{b}^{2r}_{[m]}[n] \sum_{l=0}^{N_m-1} \omega^{2^m l(\nu-n)} \\
&= \frac{1}{2^m}\sum_{n=0}^{N-1} \omega^{kn}\, \hat{b}^{2r}_{[m]}[n]\, \delta[\nu-n](\text{mod}\, N_m) \\
&= \frac{1}{2^m}\sum_{n=0}^{2^m-1} \omega^{k(\nu+N_m n)} \hat{b}^{2r}_{[m]}[\nu+N_m n] \qquad (13.6) \\
&= \frac{1}{2^m}\sum_{l=0}^{2^m-1} \omega^{k(\nu+N_m l)} \left(\frac{\sin(2^m\pi\,(\nu+N_m l))/N)}{2^m\,\sin(\pi(\nu+N_m l)/N)}\right)^{2r} \qquad (13.7) \\
&= \frac{\sin^{2r}(\pi\,\nu/N_m)}{2^m}\sum_{l=0}^{2^m-1} \frac{\omega^{k(\nu+N_m l)}}{(2^m\,\sin(\pi(\nu+N_m l)/N))^{2r}}.
\end{aligned}$$

It follows from Eq. (13.7) that the DFT of the exponential splines $\zeta^{2r}_{[m]}(\nu)$ comprises only 2^m terms

$$\hat{\zeta}^{2r}_{[m]}(\nu)[n] = N_m\, \hat{b}^{2r}_{[m]}[n]\, \delta[\nu - n](\mathrm{mod}\, N_m) \tag{13.8}$$
$$= N_m \left(\frac{\sin 2^m \pi\, n/N)}{2^m\, \sin \pi n/N)}\right)^{2r} \delta[\nu - n](\mathrm{mod}\, N_m)$$

The set of N-periodic with respect to k discrete splines $\zeta^{2r}_{[m]}(\nu) = \left\{\zeta^{2r}_{[m]}(\nu)[k]\right\}$ is N_m-periodic with respect to ν.

The discrete splines are represented via the exponential splines:

$$b^{2r}_{[m]}[k - 2^m l] = \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} \omega^{-2^m\, \nu l}\, \zeta^{2r}_{[m]}(\nu)[k]$$
$$s_{[m]}[k] = \frac{1}{N_m} \sum_{l=0}^{N_m-1} q[l] \sum_{\nu=0}^{N_m-1} \omega^{-2^m\, \nu l}\, \zeta^{2r}_{[m]}(\nu)[k] \tag{13.9}$$
$$= \frac{1}{N_m} \sum_{\nu=0}^{N_m-1} \xi_{[m]}[\nu]\, \zeta^{2r}_{[m]}(\nu)[k] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \xi_{[m]}[\nu]\, \zeta^{2r}_{[m]}(\nu)[k],$$
$$\xi_{[m]}[\nu] \stackrel{\mathrm{def}}{=} \hat{q}[n]_m = \sum_{l=0}^{N_m-1} \omega^{2^m\, nl}\, q[l].$$

Remark 13.1.2 The periodic discrete exponential splines in the limit case of the signal space $S_{[0]}$ are

$$\zeta_{[0]}(\nu)[k] = \sum_{l=0}^{N-1} \omega^{\nu l}\, b_{[0]}[k - l] = \sum_{l=0}^{N-1} \omega^{\nu l}\, \delta[k - l] = \omega^{k\nu} = e^{2\pi i \nu k/N}.$$

Thus, the representation Eq. (13.9) of a signal $\mathbf{x} = \{x[k]\} = \mathbf{s}_{[0]}$ reduces to the inverse formula for the DFT

$$x[k] = \frac{1}{N} \sum_{\nu=0}^{N-1} \hat{x}(\nu)\, \omega^{k\nu}$$

The recursive relation (13.2) between the DFT of B-splines of different spans results in a two-scale relation between the exponential splines with different spans.

Proposition 13.2 *The discrete exponential spline $\zeta^{2r}_{[m+1]}(\nu)$ can be derived from $\zeta^{2r}_{[m]}(\nu)$ by the following relation:*

$$\zeta^{2r}_{[m+1]}(\nu)[k] = \frac{1}{2}\left(b_{[m]}(\nu)\ \zeta^{2r}_{[m]}(\nu)[k] + b_{[m]}\left(\nu + \frac{N_m}{2}\right)\ \zeta^{2r}_{[m]}\left(\nu + \frac{N_m}{2}\right)[k]\right),$$
$$b_{[m]}(\nu) \stackrel{\text{def}}{=} \cos^{2r}\frac{\pi\,\nu}{N_m}. \tag{13.10}$$

Proof Equations (13.6) and (13.2) imply that

$$\zeta^{2r}_{[m+1]}(\nu)[k] = \frac{1}{2^{m+1}}\sum_{n=0}^{2^{m+1}-1}\omega^{k(\nu+N_{m+1}n)}\hat{b}^{2r}_{[m+1]}[\nu + N_{m+1}\,n]$$
$$= \frac{1}{2^{m+1}}\sum_{n=0}^{2^{m+1}-1}\omega^{k(\nu+N_{m+1}\,n)}\ \cos^{2r}\pi\ \left(\frac{\nu}{N_m} + \frac{n}{2}\right)\hat{b}^{2r}_{[m]}[\nu + N_{m+1}n]. \tag{13.11}$$

Separate even and odd terms in the sum Eq. (13.11).

$$\zeta^{2r}_{[m+1]}(\nu)[k] = \frac{1}{2}\cos^{2r}\frac{\pi\,\nu}{N_m}\,\frac{1}{2^m}\sum_{n=0}^{2^m-1}\omega^{k(\nu+N_m n)}\hat{b}^{2r}_{[m]}[\nu + N_m\,n]$$
$$+ \frac{1}{2}\sin^{2r}\frac{\pi\,\nu}{N_m}\,\frac{1}{2^m}\sum_{n=0}^{2^m-1}\omega^{k(\nu+N_m n)+N_{m+1}}\hat{b}^{2r}_{[m]}[\nu + N_m\,n + N_{m+1}]$$
$$= \frac{1}{2}\cos^{2r}\frac{\pi\,\nu}{N_m}\,\zeta^{2r}_{[m]}(\nu)[k] + \frac{1}{2}\sin^{2r}\frac{\pi\,\nu}{N_m}\,\zeta^{2r}_{[m]}(\nu + N_{m+1})[k].$$

■

The characteristic sequence of the space ${}^{2r}\mathcal{S}_{[m]}$ (the MATLAB function `per_dspUP`) is

$$U^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \zeta^{2r}_{[m]}(\nu)[0] = \sum_{l=0}^{N_m-1}\omega^{-2^m\nu l}\,b^{2r}_{[m]}[2^m l] \tag{13.12}$$
$$= \frac{1}{2^m}\sum_{n=0}^{2^m-1}\hat{b}^{2r}_{[m]}[\nu + N_m n] = \frac{1}{2^m}\sum_{n=0}^{2^m-1}\left(\frac{\sin(2^m\pi\,(\nu + N_m n))/N)}{2^m\,\sin(\pi(\nu + N_m n)/N)}\right)^{2r}$$
$$= \frac{\sin^{2r}(\pi\,\nu/N_m)}{2^m}\sum_{l=0}^{2^m-1}\frac{1}{(2^m\,\sin(\pi(\nu + N_m l)/N))^{2r}}.$$

The sequences $U^{2r}_{[m]}[\nu]$ are strictly positive for all υ. For all the splines orders, $U^{2r}_{[0]}[\nu] \equiv 1$. Equation (13.10) implies a recursive relation for the characteristic sequences $U^{2r}_{[m]}[\upsilon]$:

$$U^{2r}_{[m+1]}[\nu] = \frac{1}{2}\left(b_{[m]}(\nu)\ U^{2r}_{[m]}[\nu] + b_{[m]}\left(\nu + \frac{N_m}{2}\right)\ U^{2r}_{[m]}\left[\nu + \frac{N_m}{2}\right]\right),$$
$$= \frac{1}{2}\left(\cos^{2r}\frac{\pi\,\nu}{N_m}\ U^{2r}_{[m]}[\nu] + \sin^{2r}\frac{\pi\,\nu}{N_m}\ U^{2r}_{[m]}\left[\nu + \frac{N_m}{2}\right]\right).$$

Proposition 3.8 presents basic properties of the discrete periodic exponential splines. We specify claims of that proposition to the splines we are dealing with in this section.

Proposition 13.3 *The following properties of the discrete periodic exponential splines* $\zeta^{2r}_{[m]}(\nu)$ *hold:*

1. *The discrete exponential splines* $\left\{\zeta^{2r}_{[m]}(\nu)\right\},\ \nu = -N_m/2, \ldots, N_m/2-1$, *form an orthogonal basis of the space* ${}^{2r}\mathcal{S}_{[m]}$.
2. *Their squared norms in the space* $\Pi[N]$ *are*

$$\left\|\zeta^{2r}_{[m]}(\nu)\right\|^2 = N_m\, U^{4r}_{[m]}[\upsilon]. \tag{13.13}$$

3. *The discrete circular convolution of two discrete exponential splines is a discrete exponential spline:*

$$\zeta^{2r}_{[m]}(\nu) \circledast \zeta^{2q}_{[m]}(\mu)[k] = N_m \delta[\nu-\mu]\, \zeta^{2(r+q)}_{[m]}(\nu)[k].$$

4. *The discrete exponential splines are the eigenvectors of the circular shift operator:*

$$\zeta^{2r}_{[m]}(\nu)[k + 2^m\,\lambda] = \omega^{2^m\,\nu\lambda}\,\zeta^{2r}_{[m]}(\nu)[k]$$
$$\Longrightarrow \zeta^{2r}_{[m]}(\nu)[2^m\,\lambda] = \omega^{2^m\,\nu\lambda}\,U^{2r}_{[m]}[\nu], \quad \frac{\zeta^{2r}_{[m]}(\nu)[2^m\,\lambda]}{U^{2r}_{[m]}[\nu]} = e^{2\pi i\nu\lambda/N_m}. \tag{13.14}$$

Equation (13.14) *means that the discrete exponential splines interpolate the Fourier exponential functions.*

5. *Grid values of a discrete spline from* ${}^{2r}\mathcal{S}_{[m]}$

$$s_{[m]}[2^m k] = \frac{1}{N_m}\sum_{\nu=N_m/2}^{N_m/2-1} \xi_{[m]}[\nu]\,\zeta^{2r}_{[m]}(\nu)[2^m k] \tag{13.15}$$
$$= \frac{1}{N_m}\sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m\,nl}\,\xi_{[m]}[\nu]\,U^{2r}_{[m]}[\nu].$$

6. *Any* N_m*-periodic signal* $\mathbf{x} \in \Pi[N_m]$ *can be interpolated by a spline* $\boldsymbol{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$: $s_{[m]}[2^m\,k] = x[k],\ k = 0, \ldots, N_m - 1$, *where*

$$s_{[m]}[k] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \frac{\hat{x}[\nu]_m}{U^{2r}_{[m]}[\nu]}\, \zeta^{2r}_{[m]}(\nu)[k]. \tag{13.16}$$

The following property of discrete periodic exponential splines is similar to the differentiation property of polynomial periodic exponential splines.

Proposition 13.4 *The central finite difference of the discrete periodic exponential spline*

$$\delta^{2d}[\zeta^{2r}_{[m]}][\nu][k] = (-1)^d \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{2d} \times \begin{cases} \zeta^{2(r-d)}_{[m]}(\nu)[k], & \text{if } d < r; \\ \omega^{k\nu}\ \delta[k](\mathrm{mod} 2^m), & \text{if } d = r. \end{cases} \tag{13.17}$$

Proof We start from the second difference, $d = 1$. Equation (13.7) implies that

$$\begin{aligned} \delta^{2d}[\zeta^{2r}_{[m]}][\nu][k] &= \zeta^{2r}_{[m]}(\nu)[k+1] - 2\zeta^{2r}_{[m]}(\nu)[k] + \zeta^{2r}_{[m]}(\nu)[k-1] \\ &= \frac{\sin^{2r}(\pi\ \nu/N_m)}{2^m} \sum_{l=0}^{2^m-1} \frac{\omega^{(k+1)(\nu+N_m l)} - 2\omega^{k(\nu+N_m l)} + \omega^{(k+1)(\nu+N_m l)}}{(2^m\ \sin(\pi(\nu+N_m l)/N))^{2r}} \\ &= \frac{\sin^{2r}(\pi\ \nu/N_m)}{2^m} \sum_{l=0}^{2^m-1} \frac{\omega^{k(\nu+N_m l)}\left(e^{2\pi i(\nu+N_m l)/N} - 2 + e^{-2\pi i(\nu+N_m l)/N}\right)}{(2^m\ \sin(\pi(\nu+N_m l)/N))^{2r}} \\ &= -4\frac{\sin^{2r}(\pi\ \nu/N_m)}{2^m} \sum_{l=0}^{2^m-1} \frac{\omega^{k(\nu+N_m l)}\ \sin^2(\pi(\nu+N_m l)/N)}{(2^m\ \sin(\pi(\nu+N_m l)/N))^{2r}} \\ &= -\frac{\sin^2(\pi\ \nu/N_m)}{4^{m-1}}\ \frac{\sin^{2(r-1)}(\pi\ \nu/N_m)}{2^m} \sum_{l=0}^{2^m-1} \frac{\omega^{k(\nu+N_m l)}}{(2^m\ \sin(\pi(\nu+N_m l)/N))^{2(r-1)}} \\ &= -\left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^2 \zeta^{2(r-1)}_{[m]}(\nu)[k]. \end{aligned}$$

The relations (13.17) for $1 < d < r$ are derived by induction. In particular,

$$\delta^{2(r-1)}[\zeta^{2r}_{[m]}[\nu][k] = (-1)^{r-1} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{2(r-1)} \zeta^2_{[m]}(\nu)[k].$$

The second difference of the second order exponential spline

$$\begin{aligned} \delta^2[\zeta^2_{[m]}][\nu][k] &= -\left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^2 \frac{1}{2^m} \sum_{l=0}^{2^m-1} \omega^{k(\nu+N_m l)} \\ &= -\omega^{k\nu} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^2 \delta[k](\mathrm{mod}\, 2^m). \end{aligned}$$

■

13.1.3 Normalized Exponential Discrete Periodic Splines

The discrete exponential splines are normalized using Eq. (13.13):

$$\gamma^{2r}_{[m]}(\nu)[k] \stackrel{\text{def}}{=} \frac{\zeta^{2r}_{[m]}(\nu)[k]}{\left\|\zeta^{2r}_{[m]}(\nu)\right\|} = \frac{\zeta^{2r}_{[m]}(\nu)[k]}{\sqrt{N_m\, U^{4r}_{[m]}[\nu]}}, \tag{13.18}$$
$$\zeta^{2r}_{[m]}(\nu)[k] = \sqrt{N_m\, U^{4r}_{[m]}[\nu]}\, \gamma^{2r}_{[m]}(\nu)[k].$$

It follows from Eq. (13.8) that the DFT of the exponential splines $\gamma^{2r}_{[m]}(\nu)$ comprises only 2^m terms

$$\hat{\gamma}^{2r}_{[m]}(\nu)[n] = \sqrt{\frac{N_m}{U^{4r}_{[m]}[\nu]}}\, \left(\frac{\sin 2^m \pi\, n/N)}{2^m\, \sin \pi n/N)}\right)^{2r}\, \delta[\nu - n](\text{mod } N_m)$$

The exponential splines $\left\{\gamma^{2r}_{[m]}(\nu)\right\}$, $n = -N_m/2, \ldots, N_m/2 - 1$, form an orthonormal basis of the space ${}^{2r}\mathscr{S}_{[m]}$. If a spline $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ is

$$s_{[m]}[k] = \sum_{l=0}^{N_m-1} q[l]\, b^{2r}_{[m]}[k - 2^m l] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \xi_{[m]}[\nu]\, \zeta^{2r}_{[m]}(\nu)[k],$$

where $\xi_{[m]}[\nu] \stackrel{\text{def}}{=} \hat{\mathbf{q}}[\nu]_m$, then it can be represented as

$$s_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k], \tag{13.19}$$
$$\sigma_{[m]}[\nu] = \sqrt{N_m}\left\langle s_{[m]}, \gamma^{2r}_{[m]}(\nu)\right\rangle = \sqrt{U^{4r}_{[m]}[\nu]}\, \hat{\mathbf{q}}[\nu]_m.$$

In particular, the B-spline

$$b^{2r}_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sqrt{U^{4r}_{[m]}[\nu]}\, \gamma^{2r}_{[m]}(\nu)[k]. \tag{13.20}$$

Properties of the normalized discrete periodic exponential splines $\gamma^{2r}_{[m]}(\nu)$ are readily derived from the properties of the exponential splines $\zeta^{2r}_{[m]}(\nu)$, which are presented in Propositions 13.3 and 13.4.

Finite differences: The central finite difference of a normalized discrete periodic exponential spline is

$$\delta^{2d}[\gamma^{2r}_{[m]}][\nu][k] = \frac{(-1)^d}{\sqrt{U^{4r}_{[m]}[\nu]}} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{2d} \times \begin{cases} \sqrt{U^{4(r-d)}_{[m]}[\nu]}\,\gamma^{2(r-d)}_{[m]}(\nu)[k], & \text{if } d < r; \\ \omega^{k\nu}\ \delta[k](\mathrm{mod}\,2^m)/\sqrt{N_m}, & \text{if } d = r. \end{cases} \tag{13.21}$$

Norms of differences:

$$\left\|\delta^{2d}[\gamma^{2r}_{[m]}][\nu]\right\|^2 = W^{r,d}_{[m]}[\nu],$$

$$W^{r,d}_{[m]}[\nu] \stackrel{\text{def}}{=} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{4d} \frac{U^{4(r-d)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}. \tag{13.22}$$

Circular shift:

$$\gamma^{2r}_{[m]}(\nu)[k + 2^m\,\lambda] = \omega^{2^m\,\nu\lambda}\,\gamma^{2r}_{[m]}(\nu)[k] \tag{13.23}$$

$$\Longrightarrow \gamma^{2r}_{[m]}(\nu)[2^m\,\lambda] = \frac{\omega^{2^m\,\nu\lambda}}{\sqrt{N_m}}\,V^{2r}_{[m]}[\nu], \quad V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}}, \quad \lambda \in \mathbb{N}.$$

Circular discrete convolution:

$$\gamma^{2r}_{[m]}(\nu) \circledast \gamma^{2q}_{[m]}(\mu)[k] = \sum_{l=0}^{N-1} \gamma^{2r}_{[m]}(\nu)[k-l]\,\gamma^{2q}_{[m]}(\mu)[l] \tag{13.24}$$

$$= \delta[\nu-\mu]\,\sqrt{N_m}\,R^{r,q}_{[m]}(\nu)\,\gamma^{2(r+q)}_{[m]}(\nu)[k],$$

$$R^{p,q}[\nu] \stackrel{\text{def}}{=} \sqrt{\frac{U^{4(r+q)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]\,U^{4q}_{[m]}[\nu]}}.$$

Remark 13.1.3 Recall that $\zeta_{[0]}(\nu)[k] = e^{2\pi i\nu k/N}$. Obviously, the squared norm

$$\left\|\zeta_{[0]}(\nu)\right\|^2 = \sum_{k=0}^{N-1} \left|\zeta_{[0]}(\nu)[k]\right|^2 = N.$$

Thus, $\gamma_{[0]}(\nu)[k] = N^{-1/2}\,e^{2\pi i\nu k/N}$ and the representation Eq. (13.19) of a signal $\mathbf{x} = \{x[k]\} = \mathbf{s}_{[0]}$ reduces to the normalized inverse DFT formula

$$x[k] = \frac{1}{\sqrt{N}} \sum_{\nu=-N/2}^{N/2-1} \bar{\hat{x}}[\nu]\,\omega^{k\nu}, \quad \bar{\hat{x}}[\nu] \stackrel{\text{def}}{=} \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x[k]\,\omega^{-k\nu}.$$

13.1.4 Representation of Discrete Periodic Splines by Exponential Splines Basis

Keeping in mind Remark 13.1.3, we may claim that expansion of discrete periodic splines over the orthonormal exponential splines basis as presented in Eq. (13.19) is a natural extension of the discrete Fourier analysis. On the other hand, it can be regarded as a discrete version of the § Harmonic Analysis (SHA), which was presented in Sect. 4.1. The discrete counterpart of the SHA-spectrum of a spline is the set of the normalized DFT coefficients $\bar{\bar{\mathbf{s}}}[\nu]$ of a discrete spline $\mathbf{s}$. Surely, basic relations in the discrete SHA are similar to the corresponding relations in the polynomial SHA.

Finite differences: The central finite difference of order $2d < 2r$ of a discrete spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ is a discrete spline from ${}^{2(r-d)}\mathcal{S}_{[m]}$. Let $\mathbf{s}_{[m]}$ be represented as in Eq. (13.19). Then, Eq. (13.21) implies for $d < r$

$$\delta^{2d}[\mathbf{s}_{[m]}][k] = \frac{(-1)^d}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} B^{r,d}_{[m]}[\nu]\, \sigma_{[m]}[\nu]\, \gamma^{2(r-d)}_{[m]}(\nu)[k],$$

$$B^{r,d}_{[m]}[\nu] \stackrel{\text{def}}{=} \sqrt{\frac{U^{4(r-d)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{2d}. \tag{13.25}$$

For $d = r$,

$$\delta^{2r}[\mathbf{s}_{[m]}][k] = \frac{(-1)^r}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} B^{r,r}_{[m]}[\nu]\, \sigma_{[m]}[\nu]\, \omega^{k\nu}\ \delta[k](\mathrm{mod}\, 2^m),$$

$$B^{r,r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{1}{\sqrt{N_m\, U^{4r}_{[m]}[\nu]}} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{2r}. \tag{13.26}$$

Parseval identities: Let $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ be represented as in Eq. (13.19) and a spline $\breve{\mathbf{s}}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ be represented as

$$\breve{s}_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \breve{\sigma}_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k]. \tag{13.27}$$

From Eqs. (13.19), (13.27) and (13.22), we get:

$$\left\langle \mathbf{s}_{[m]}, \breve{\mathbf{s}}_{[m]} \right\rangle = \sum_{k=0}^{N-1} s_{[m]}[k]\ \breve{s}^*_{[m]}[k]$$

$$= \frac{1}{N_m} \sum_{\nu,\mu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \breve{\sigma}^*_{[m]}[\mu] \sum_{k=0}^{N-1} \gamma^{2r}_{[m]}(\nu)[k]\, \gamma^{2r}_{[m]}(\mu)[k]^*$$

$$= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \breve{\sigma}^*_{[m]}[\nu] \sum_{k=0}^{N-1} \left|\gamma^{2r}_{[m]}(\nu)[k]\right|^2$$

$$= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \breve{\sigma}^*_{[m]}[\nu] \tag{13.28}$$

$$\|\mathbf{s}_{[m]}\|^2 = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} |\sigma_{[m]}[\nu]|^2,$$

$$\left\|\delta^{2d}[\mathbf{s}_{[m]}]\right\|^2 = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} |\sigma_{[m]}[\nu]|^2\ \breve{W}^{r,d}_{[m]}[\nu] \tag{13.29}$$

$$\text{where} \breve{W}^{r,d}_{[m]}[\nu] \stackrel{\text{def}}{=} \left(\frac{2\sin(2^m\pi\ \nu/N)}{2^m}\right)^{4d}\ \frac{U^{4(r-d)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}.$$

Circular convolution: Circular discrete convolution of two discrete splines is a discrete spline, whose order is equal to the sum of the convolved splines' orders. The respective coordinates are multiplied. Let $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ be represented as in Eq. (13.19) and the spline $\breve{\mathbf{s}}_{[m]} \in {}^{2q}\mathscr{S}_{[m]}$ be represented as

$$\breve{s}_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \breve{\sigma}_{[m]}[\nu]\, \gamma^{2q}_{[m]}(\nu)[k]. \tag{13.30}$$

From Eq. (13.24), we have

$$\mathbf{s}_{[m]} \circledast \breve{s}_{[m]}[k] = \sum_{l=0}^{N-1} \mathbf{s}_{[m]}[k-l]\, \breve{s}_{[m]}[l]$$

$$= \frac{1}{N_m} \sum_{\nu,\mu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \breve{\sigma}_{[m]}[\mu]\, \gamma^{2r}_{[m]}(\nu)[k-l] \circledast \gamma^{2r}_{[m]}(\mu)[l]$$

$$= \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \breve{\sigma}_{[m]}[\nu]\ R^{r,q}_{[m]}(\nu)\ \gamma^{2(r+q)}_{[m]}(\nu)[k] \in {}^{2(r+q)}\mathscr{S}_{[m]}. \tag{13.31}$$

Sampling and interpolation: If a spline $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ is expanded over the orthonormal basis as in Eq. (13.19) then, modifying Eq. (13.15), we present its grid samples as follows:

$$s_{[m]}[2^m l] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m \nu l}\, \sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu], \tag{13.32}$$

$$V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}}.$$

Equation (13.32) means that the samples $\left\{s_{[m]}[2^m k]\right\}$, $k = 0, \ldots, N-1$ are produced by the inverse DFT of the sequence $\left\{\sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu]\right\}$, $n = -N_m/2, \ldots, N_m/2 - 1$.

Assume the spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ interpolates an N_m-periodic signal $\mathbf{x} \in \Pi[N_m]$: $s_{[m]}[2^m l] = x[k]$, $k = 0, \ldots, N_m - 1$. Then, modifying Eq. (13.16), we represent the spline

$$s_{[m]}[k] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k], \quad \sigma_{[m]}[\nu] = \frac{\hat{\mathbf{x}}[\nu]}{V^{2r}_{[m]}[\nu]}. \tag{13.33}$$

Sampled Parseval identity: It follows from Eqs. (1.24) and (13.32)

$$\sum_{k=0}^{N_m-1} \left|\mathbf{s}_{[m]}[2^m k]\right|^2 = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \left|\sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu]\right|^2. \tag{13.34}$$

Sampled circular convolution: Let $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ and $\check{\mathbf{s}}_{[m]} \in {}^{2q}\mathcal{S}_{[m]}$ be represented as in Eqs. (13.19) and (13.30), respectively. Then, substituting $k = 2^m \kappa$ into Eq. (13.31), we get

$$\begin{aligned}
\mathbf{s}_{[m]} \circledast \check{\mathbf{s}}_{[m]}[2^m \kappa] &= \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \check{\sigma}_{[m]}[\nu]\, \bar{R}^{r,q}_{[m]}(\nu)\, \gamma^{2(r+q)}_{[m]}(\nu)[2^m \kappa] \\
&= \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \omega^{2^m \kappa \nu}\, Q^{r,q}_{[m]}[\nu]\, \sigma_{[m]}[\nu]\, \check{\sigma}_{[m]}[\nu]\,, \\
Q^{r,q}_{[m]}[\nu] &\stackrel{\text{def}}{=} \frac{U^{2(r+q)}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]\, U^{4r}_{[m]}[\nu]}}.
\end{aligned}$$

13.1.4.1 Explicit Computation of a Periodic Discrete Spline

A significant advantage of periodic discrete splines is the fact that their values can be explicitly calculated using the DFT. Assume a spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ is represented via the orthonormal basis as in Eq. (13.19)

$$s_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k].$$

An alternative representation is via the circular shifts of the discrete B-spline as in Eq. (13.3)

$$s_{[m]}[k] = \sum_{l=0}^{N_m-1} q[l]\, b^{2r}_{[m]}[k - 2^m l],$$

where the vector $\mathbf{q} = \{q[k]\}$ is an N_m-periodic sequence, which is linked to the coordinates $\sigma_{[m]}[n]$ via the DFT, as is indicated in Eq. (13.19):

$$\hat{\mathbf{q}}[n]_m = \sum_{k=0}^{N_m-1} \omega^{-2^m kn} q[k] = \frac{\sigma_{[m]}[n]}{\sqrt{U^{4r}_{[m]}[n]}}.$$

Then, Eq. (13.4) implies that

$$\hat{\mathbf{s}}^{2r}_{[m]}[n] = \sum_{k=0}^{N-1} \omega^{-nk}\, s^{2r}_{[m]}[k] = \hat{q}[n]_m \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)}\right)^{2r}$$

$$\Longrightarrow s^{2r}_{[m]}[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \frac{\sigma_{[m]}[n]}{\sqrt{U^{4r}_{[m]}[n]}} \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)}\right)^{2r}. \tag{13.35}$$

In particular, assume, grid samples of the spline $\mathbf{s}_{[m]}$ are available: $s_{[m]}[2^m k] = x[k]$. Then, due to Eqs. (13.32) and (13.33),

$$\sigma_{[m]}[n] = \frac{\hat{\mathbf{x}}[n] \sqrt{U^{4r}_{[m]}[n]}}{U^{2r}_{[m]}[n]}$$

$$\Longrightarrow s^{2r}_{[m]}[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \frac{\hat{\mathbf{x}}[n]_m}{U^{2r}_{[m]}[n]} \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)}\right)^{2r}.$$

These values are readily calculated by application of the inverse FFT (IFFT).

13.1.5 Generators of Periodic Discrete Splines' Spaces

An N-periodic spline $\varphi_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ is a generator of the discrete periodic splines' space ${}^{2r}\mathscr{S}_{[m]}$ if any spline from $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ can be represented as a linear combination

$$s_{[m]}[k] = \sum_{l=0}^{N_m-1} r[l]\,\varphi_{[m]}[k - 2^m\,l], \quad N_m = N\,2^m.$$

An obvious generator of ${}^{2r}\mathscr{S}_{[m]}$ is the periodic discrete B-spline $\mathbf{b}^{2r}_{[m]}$. Any spline, including candidates for generators, is represented as

$$\varphi_{[m]}[k] = \sum_{l=0}^{N_m-1} f[k]\,b^{2r}_{[m]}[k - 2^m\,l] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \tau[\nu]\,\gamma^{2r}_{[m]}(\nu)[k], \tag{13.36}$$

$$\tau[\nu] = \sqrt{N_m\,U^{4r}_{[m]}[\nu]}\;\hat{f}[n]_m, \quad \hat{f}[n]_m \stackrel{\text{def}}{=} \sum_{k=-N_m/2}^{N_m/2-1} \omega^{-2^m kn}\,f[k].$$

Equation (13.23) implies that

$$\varphi_{[m]}[k - 2^m\,l] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \tau[\nu]\,\gamma^{2r}_{[m]}(\nu)[k - 2^m\,l] \tag{13.37}$$

$$= \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{-2^m\,l\nu}\tau[\nu]\,\gamma^{2r}_{[m]}(\nu)[k],$$

$$\Longrightarrow \tau[\nu]\,\gamma^{2r}_{[m]}(\nu)[k] = \frac{1}{\sqrt{N_m}} \sum_{l=0}^{N_m-1} \omega^{2^m\,l\nu}\,\varphi_{[m]}[k - 2^m\,l]. \tag{13.38}$$

There holds a condition for a spline to be a generator. The condition is similar to Proposition 4.4.

Proposition 13.5 *A discrete spline* $\varphi_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ *represented by Eq.* (13.36) *can serve as a generator of the space* ${}^{2r}\mathscr{S}_{[m]}$ *if its coordinates* $\{\tau[\nu]\}$, $n = -N_m/2, \ldots, N_m/2 - 1$, *do not vanish.*

Proof Let a spline $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ be represented as

$$s_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \sigma[\nu]\,\gamma^{2r}_{[m]}[\nu][k].$$

Then, using Eq. (13.38) we can write

$$s_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \frac{\sigma[\nu]}{\tau[\nu]}\,\tau[\nu]\,\gamma^{2r}_{[m]}[\nu][k]$$

$$= \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \frac{\sigma[n]}{\tau[n]} \sum_{l=0}^{N_m-1} \omega^{2^m l\nu}\, \varphi_{[m]}[k-2^m l]$$
$$= \sum_{l=0}^{N_m-1} r[l]\, \varphi_{[m]}[k-2^m l], \quad r[l] = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \omega^{2^m l\nu}\, \frac{\sigma[n]}{\tau[n]}.$$

■

Proposition 13.6 *Assume a spline $\varphi_{[m]}$ presented as in Eq.* (13.36) *is a generator of ${}^{2r}\mathscr{S}_{[m]}$. Then there exists a unique dual generator*

$$\phi_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \theta[\nu]\, \gamma_{[m]}^{2r}(\nu)[k]$$

such that the biorthogonal relations

$$\sum_{k=0}^{N-1} \varphi_{[m]}[k-2^m l]\, \phi^*_{[m]}[k-2^m r] = \delta[l-r].$$

The coordinates $\{\tau[\nu]\}$ and $\{\theta[\nu]\}$ of the generators $\varphi_{[m]}$ and $\phi_{[m]}$, respectively, are linked as

$$\tau[\nu]\, \theta[\nu]^* = 1. \tag{13.39}$$

Proof Equation (13.37) implies that the splines $\varphi_{[m]}[k-2^m l]$ and $\phi_{[m]}[k-2^m r]$ are

$$\varphi_{[m]}[k-2^m l] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{-2^m l\nu} \tau[\nu]\, \gamma_{[m]}^{2r}(\nu)[k],$$
$$\phi_{[m]}[k-2^m r] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{-2^m r\nu} \theta[\nu]\, \gamma_{[m]}^{2r}(\nu)[k].$$

Then, using the Parseval identity (13.28), and Eq. (13.39), we get

$$\sum_{k=0}^{N-1} \varphi_{[m]}[k-2^m l]\, \phi^*_{[m]}[k-2^m r] = \delta[l-r]$$
$$= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \tau[\nu]\, \theta^*[\nu] \omega^{-\nu 2^m (l-r)} = \delta[l-r].$$

■

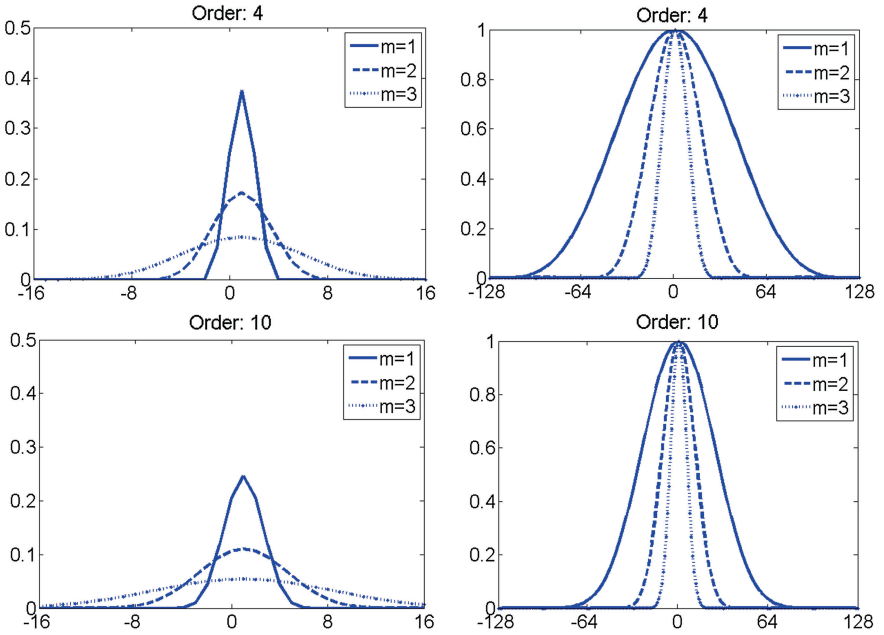


Fig. 13.1 *Top left* discrete 256-periodic B-splines $\mathbf{b}^4_{[m]}$ of orders 4 for $m = 1, 2, 3$; *Top right* their DFT spectra; *Bottom* the same for the B-splines $\mathbf{b}^{10}_{[m]}$ of order 10

13.1.5.1 Examples of Generators of the Periodic Splines Spaces

All the figures in this section are produced by the MATLAB code `per_dsp_gen_exampP`.

Periodic discrete B-splines: Due to Eq. (13.1), values of the discrete periodic B-spline

$$b^{2r}_{[m]}[k] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{kn} \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)} \right)^{2r}$$

are calculated via the IDFT. Figure 13.1 displays the 256-periodic discrete B-splines of orders 4 and 10 for the scales $m = 1, 2, 3$ and their DFT spectra. Visually, shapes of the B-splines and their DFT spectra are similar to those of the polynomial B-splines.

Generators dual to discrete periodic B-splines: Expansion of the discrete B-spline (Eq. 13.20) is

$$b^{2r}_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sqrt{U^{4r}_{[m]}[\nu]}\, \gamma^{2r}_{[m]}(\nu)[k].$$

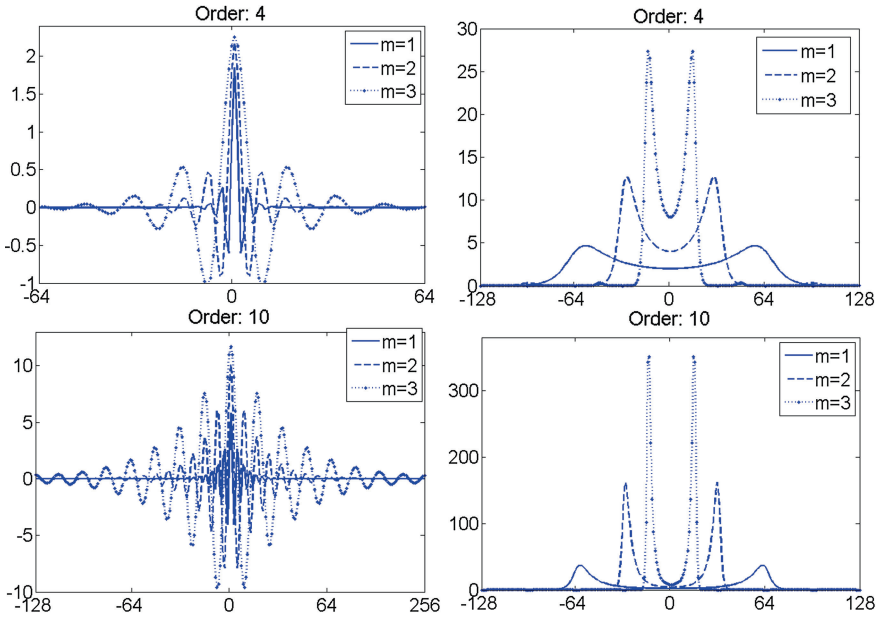


Fig. 13.2 *Top left* discrete 256-periodic splines $\mathbf{d}^4_{[m]}$ of orders 4 for $m = 1, 2, 3$; *Top right* their DFT spectra; *Bottom* the same for the splines $\mathbf{d}^{10}_{[m]}$ of order 10

Then, Eq. (13.39) implies that the discrete spline

$$d^{2r}_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \frac{\gamma^{2r}_{[m]}(\nu)[k]}{\sqrt{U^{4r}_{[m]}[\nu]}}$$

is dual to the B-spline $\mathbf{b}^{2r}_{[m]}$. Equation (13.35) provides an explicit expression for the splines $\mathbf{d}^{2r}_{[m]}$ via the IDFT

$$d^{2r}_{[m]}[k] = \frac{1}{N} \sum_{k=0}^{N-1} \omega^{nk} \frac{1}{U^{4r}_{[m]}[\nu]} \left(\frac{\sin(2^m \pi\, \nu/N)}{2^m \sin(\pi\, \nu/N)} \right)^{2r}.$$

Figure 13.2 displays the 256-periodic discrete splines dual to B-splines of orders 4 and 10 for $m = 1, 2, 3$ and their DFT spectra.

The discrete periodic generator whose shifts are orthogonal to each other (self-dual generator):

It is readily seen from Eq. (13.39) that the coordinates of such a spline $\varphi^{2r}_{[m]}$ should be $\tau[\nu] = 1,\ n = -N_m/2, \ldots, N_m/2 - 1$. Thus,

$$\varphi^{2r}_{[m]}[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \omega^{\nu k} \frac{1}{\sqrt{U^{4r}_{[m]}[\nu]}} \left(\frac{\sin(2^m \pi\, \nu/N)}{2^m \sin(\pi\, \nu/N)} \right)^{2r}. \tag{13.40}$$

The splines $\varphi^{2r}_{[m]}[\cdot - 2^m l],\ l = 0, \ldots N_m - 1$, form an orthonormal basis of the space ${}^{2r}\mathcal{S}_{[m]}$. Any spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ can be expanded as

$$s_{[m]}[k] = \sum_{l=0}^{N_m-1} s[l]\, \varphi^{2r}_{[m]}[k - 2^m\, l],$$

$$s[l] = \left\langle \mathbf{s}_{[m]}, \varphi^{2r}_{[m]}[\cdot - 2^m l] \right\rangle = \sum_{k=0}^{N-1} s_{[m]}[k]\, \varphi^{2r}_{[m]}[k - 2^m\, l].$$

Figure 13.3 displays the 256-periodic self-dual discrete splines of orders 4 and 10 for $m = 1, 2, 3$ and their DFT spectra. We observe that, although the splines supported on the whole interval $N/2 \geq k < N/2$, they are well localized. Their DFT spectra, like the SHA spectra of their polynomial counterparts, have near a rectangular shape.

The discrete splines space ${}^{2r}\mathcal{S}_{[m]}$ is a subspace of the space $\Pi[N]$ of N-periodic signals. The orthogonal projection of the space $\Pi[N]$ onto the subspace ${}^{2r}\mathcal{S}_{[m]}$ can be expressed explicitly.

Proposition 13.7 *Assume a signal* $\mathbf{x} = \{x[k]\}$ *belongs to the space* $\Pi[N]$. *Then, its orthogonal projection onto the subspace* ${}^{2r}\mathcal{S}_{[m]}$

$$\begin{aligned} x_{[m]}[k] &= \sum_{l=0}^{N_m-1} \xi_{[m]}[l]\, \varphi^{2r}_{[m]}[k - 2^m\, l], \quad \xi_{[m]}[l] = \left\langle \mathbf{x}, \varphi^{2r}_{[m]}[\cdot - 2^m l] \right\rangle \qquad (13.41) \\ &= \sum_{k=0}^{N-1} x[k]\, \varphi^{2r}_{[m]}[k - 2^m\, l] \\ &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{2^m\, ln} \frac{\hat{x}[n]}{\sqrt{U^{4r}_{[m]}[n]}} \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)} \right)^{2r}. \end{aligned}$$

It is a direct consequence of orthonormality of the splines $\left\{ \varphi^{2r}_{[m]}[\cdot - 2^m l] \right\},\ l = 0, \ldots, N_m - 1$, and the representation Eq. (13.40).

Remark 13.1.4 Equation (13.41) offers a direct way to compute coefficients of the orthogonal projection spline $\mathbf{x}_{[m]}$. That is to calculate the IDFT

$$\eta[k] = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{kn} \frac{\hat{x}[n]}{\sqrt{U^{4r}_{[m]}[n]}} \left(\frac{\sin(2^m \pi\, n/N)}{2^m \sin(\pi\, n/N)} \right)^{2r}$$

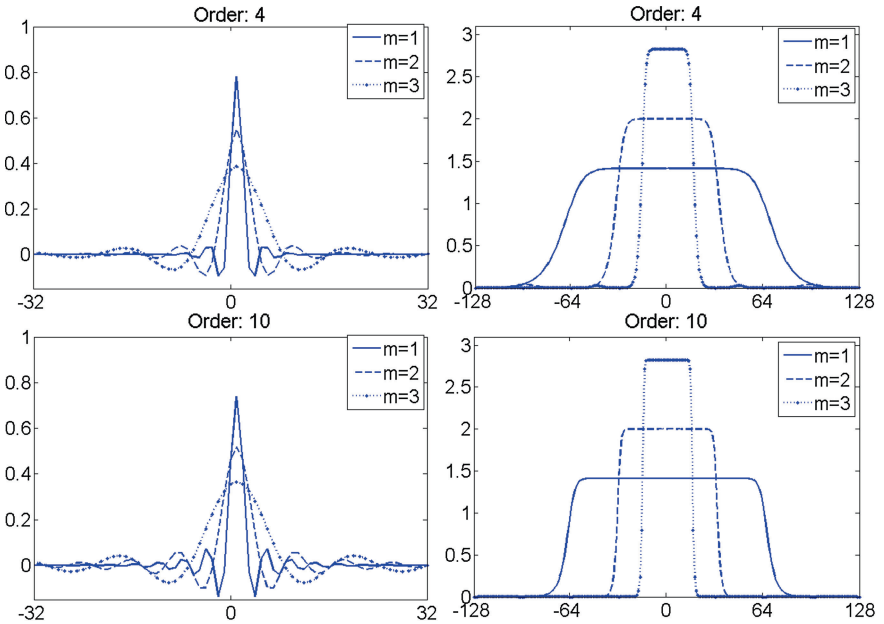


Fig. 13.3 *Top left* discrete 256-periodic splines $\varphi^4_{[m]}$ of orders 4 for $m = 1, 2, 3$; *Top right* their DFT spectra; *Bottom* the same for the splines $\varphi^{10}_{[m]}$ of order 10

and to downsample it: $\xi_{[m]}[l] = \eta[2^m\, l],\ l = 0, \ldots, N_m - 1$.

In Sect. 14.1 we discuss an algorithm for that purpose. The algorithm is based on two-scale relations.

The interpolating generators (fundamental discrete periodic splines): Define the so-called fundamental spline $\mathbf{l}^{2r}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$

$$l^{2r}_{[m]}[k] \stackrel{\text{def}}{=} \frac{1}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \frac{\gamma^{2r}_{[m]}(\nu)[k]}{V^{2r}_{[m]}[\nu]}, \quad V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}}$$

$$= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \omega^{nk}\, \frac{1}{U^{2r}_{[m]}[n]} \left(\frac{\sin(2^m \pi\, n/N)}{2^m\, \sin(\pi\, n/N)} \right)^{2r}.$$

It follows from Eq. (13.23) that values of the spline at grid points

$$l^{2r}_{[m]}[2^m n] = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \omega^{2^m n \nu} = \delta[k] (\mathrm{mod} N_m).$$

The fundamental spline provides an explicit representation for the spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$, which interpolates a signal $\mathbf{z} = \{z[k]\} \in \Pi[N_m]$ at grid points:

$$\begin{aligned} s_{[m]}[k] &= \sum_{l=0}^{N_m-1} z[l]\, l^{2r}_{[m]}[k - 2^m l] \\ &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} w^{nk}\, \frac{\hat{z}[n]}{U^{2r}_{[m]}[n]} \left(\frac{\sin(2^m \pi\, n/N)}{2^m\, \sin(\pi\, n/N)} \right)^{2r}, \qquad (13.42) \\ s_{[m]}[2^m n] &= \sum_{l=0}^{N_m-1} z[l]\, l^{2r}_{[m]}[2^m (n-l)] = z[n]. \end{aligned}$$

The discrete interpolatory splines possess an extremal property similar to the extremal property of polynomial splines, which presented in Proposition 5.1. Denote, by Π_z the subspace of signals $\mathbf{y} \in \Pi[N]$ such that $y[2^m k] = z[k],\ k = 0, \ldots, N_m - 1$.

Proposition 13.8 ([2, 1]) *The discrete spline* $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$ *defined by Eq.* (13.42) *provides the unique solution to the extremal problem*

$$I(\mathbf{y}) \stackrel{\text{def}}{=} \left\| \Delta^r[\mathbf{y}] \right\|^2_{\mathbf{y} \in \Pi_z} \longrightarrow \min .$$

Here, $\Delta^r[\mathbf{y}]$ means the finite difference of order r.

Remark 13.1.5 Loosely speaking, the discrete spline, which interpolates the N_m-periodic signal $\mathbf{z}$ is the smoothest of all the N-periodic interpolatants. Thus, Eq. (13.42) provides a smooth upsampling of the signal $\mathbf{z}$. When the signal $\mathbf{z}$ is distorted by noise, utilization of the so-called *smoothing* discrete splines to be discussed in Sect. 13.1.6 is a good choice.

Figure 13.4 displays the 256-periodic fundamental discrete splines of orders 4 and 10 for $m = 1, 2, 3$ and their DFT spectra.

13.1.6 Smoothing Periodic Discrete Splines

Design of discrete splines, which smooth noised data is similar to design of polynomial smoothing splines. The idea about how to smooth noised sequences is due to Whittaker [3]. Assume that samples $x[2^m l],\ l = 0, \ldots, N_m$, of a signal $\mathbf{x} \in \Pi[N]$ are corrupted by a Gaussian zero mean noise $\mathbf{e} = \{e_l\},\ l = 0, \ldots, N_m$, whose variance $\mathrm{var}(\mathbf{e}) = \varepsilon^2$. Thus, we approximate the signal $\mathbf{x} \in \Pi[N]$ by a signal $\mathbf{y} \in \Pi[N]$ derived from the sparse corrupted data $\mathbf{z} = \{z[l] = x[2^m l] + e_l\},\ l = 0, \ldots, N_m$.

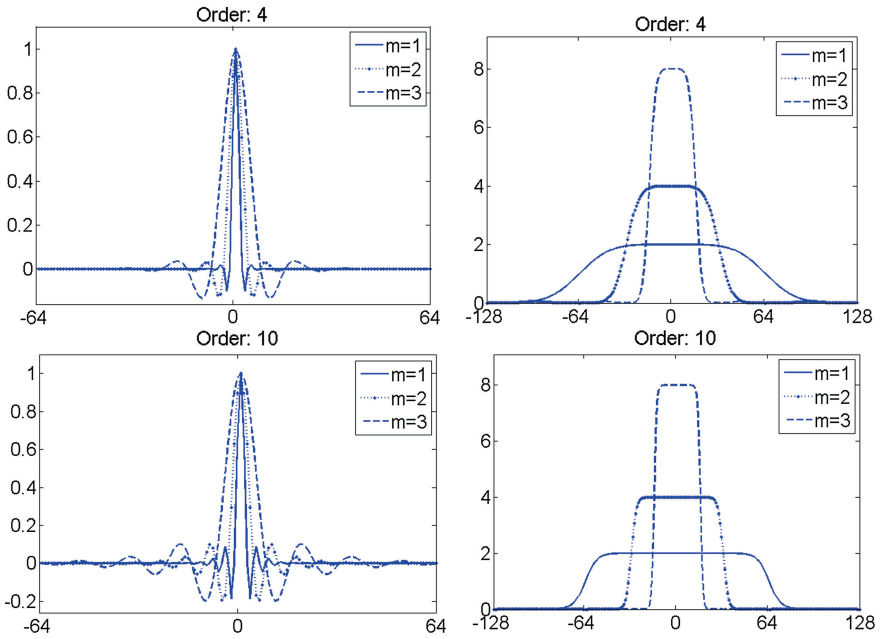


Fig. 13.4 *Top left* discrete 256-periodic splines $\mathbf{l}^4_{[m]}$ of orders 4 for $m = 1, 2, 3$; *Top right* their DFT spectra; *Bottom* the same for the splines $\mathbf{l}^{10}_{[m]}$ of order 10

13.1.6.1 Parameterized Discrete Splines

As in continuous setting, we look for a signal $\mathbf{y}(\rho) \in \Pi[N]$, which minimizes the parameterized functional $J_\rho(\mathbf{y}) \stackrel{\text{def}}{=} \rho I(\mathbf{y}) + E(\mathbf{y})$, where

$$I(\mathbf{y}) \stackrel{\text{def}}{=} \left\| \Delta^r[\mathbf{y}] \right\|^2, \quad E(\mathbf{y}) \stackrel{\text{def}}{=} \sum_{l=0}^{N_m-1} (y[2^m l] - z[l])^2.$$

The regularization parameter ρ provides a trade-off between regularity of the solution (functional $I(\mathbf{y})$) and approximation of the available data $\mathbf{z}$ (functional $E(\mathbf{y})$). When $\rho = 0$, the minimum of $J_\rho(\mathbf{y})$ is achieved on the discrete spline $\mathbf{s}_{[m]} \in {}^{2r}\mathcal{S}_{[m]}$, which interpolates the data $\mathbf{z}$ (Proposition 13.8). A similar fact holds for non-zero ρ.

Proposition 13.9 ([1, 2]) *A unique solution to the unconstrained minimization problem* $\min_{\mathbf{y}\in\Pi[N]} J_\rho(\mathbf{y})$ *is a discrete spline* $\mathbf{s}_{[m]}(\rho) \in {}^{2r}\mathcal{S}_{[m]}$ *of order* $2r$.

The spline $\mathbf{s}_{[m]}(\rho) \in {}^{2r}\mathcal{S}_{[m]}$, which minimizes the functional J_ρ is called the discrete periodic smoothing spline.

For simplicity, we assume that $r = 2d$ is an even number. Then, due to periodicity of signals the functional

$$I(\mathbf{y}) \stackrel{\text{def}}{=} \left\| \Delta^{2d}[\mathbf{y}] \right\|^2 = \left\| \delta^{2d}[\mathbf{y}] \right\|^2,$$

where $\delta^{2d}[\mathbf{y}]$ is the central difference of even order r. Assume that a spline $\mathbf{s}_{[m]} \in {}^{2r}\mathscr{S}_{[m]}$ is represented via the orthonormal basis as in Eq. (13.19)

$$s_{[m]}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k].$$

Then, Eq. (13.29) implies that

$$I(\mathbf{s}_{[m]}) = \left\| \delta^{2d}[\mathbf{s}_{[m]}] \right\|^2 = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} |\sigma_{[m]}[\nu]|^2\, W^{r,r}_{[m]}[\nu] \tag{13.43}$$

where

$$W^{r,r}_{[m]}[\nu] \stackrel{\text{def}}{=} \left(\frac{2\sin(2^m \pi\ \nu/N)}{2^m} \right)^{2r} \frac{U^{2r}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}.$$

Equation (13.32) provides grid samples of the spline

$$s_{[m]}[2^m l] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m\, \nu l}\, \sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu], \quad V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}},$$

which are represented via the IDFT, as well as the data vector:

$$z[l] = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m\, \nu l}\, \hat{z}[\nu]_m.$$

Then, using the Parseval identity Eq. (1.24), we have

$$E(\mathbf{s}_{[m]}) \stackrel{\text{def}}{=} \sum_{l=0}^{N_m-1} (s_{[m]}[2^m\, l] - z[l])^2 = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \left| \sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu] - \hat{z}[\nu]_m \right|^2. \tag{13.44}$$

Hence, we obtain an explicit expression of the parameterized functional

$$J_\rho(\mathbf{s}_{[m]}) = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \left(\rho\, |\sigma_{[m]}[\nu]|^2\, W^{r,r}_{[m]}[\nu] + \left| \sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu] - \hat{z}[\nu]_m \right|^2 \right).$$

This expression is quite similar to the expression Eq. (5.5) of the parameterized functional in the polynomial splines case.

Proposition 13.10 *Minimum to the functional* $J_\rho(\mathbf{s}_{[\mathbf{m}]})$ *on the discrete spline space* ${}^{2r}\mathcal{S}_{[m]}$ *is provided by the spline*

$$s_{[m]}(\rho)[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\nu]\, \gamma^{2r}_{[m]}(\nu)[k], \tag{13.45}$$

$$\sigma_{[m]}[\nu] = \frac{V^{2r}_{[m]}[\nu]\, \hat{z}[\nu]_m}{\rho\, W^{r,r}_{[m]}[\nu] + (V^{2r}_{[m]}[\nu])^2} = \frac{\sqrt{U^{4r}_{[m]}[\nu]}\, \hat{z}[\nu]_m}{\bar{A}^r_\rho[\nu]},$$

$$\bar{A}^r_\rho[\nu] \stackrel{\text{def}}{=} \rho \left(2^{1-m} \sin \frac{2^m \pi\, \nu}{N}\right)^{2r} + U^{2r}_{[m]}[\nu].$$

The proof of this statement is similar to the proof of Proposition 5.3.

Using Eq. (13.35), we derive an explicit expression of the smoothing spline:

$$s_{[m]}(\rho)[k] = \frac{1}{N} \sum_{k=0}^{N-1} \omega^{nk}\, \frac{\hat{z}[\nu]_m}{\bar{A}^r_\rho[\nu]} \left(\frac{\sin(2^m \pi\ \nu/N)}{2^m\ \sin(\pi\ \nu/N)}\right)^{2r}.$$

Grid samples of the spline

$$\begin{aligned} s_{[m]}(\rho)[2^m l] &= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m\, \nu l}\, \sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu] \\ &= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \omega^{2^m\, \nu l}\, \frac{U^{2r}_{[m]}[\nu]\, \hat{z}[\nu]_m}{\bar{A}^r_\rho[\nu]} \end{aligned}$$

Obviously, when $\rho = 0$, the spline $\mathbf{s}_{[m]}(\rho)$ turns into an interpolatory one.

13.1.6.2 Optimal Regularization Parameter

The scheme of selection of the parameter ρ, which provides an optimal trade-off between regularity of the spline $\mathbf{s}_{[m]}(\rho)$ and approximation of the available data $\mathbf{z}$, is similar to the scheme applied to the continuous splines.

Denote, $e(\rho) \stackrel{\text{def}}{=} E(\mathbf{s}_{[m]}(\rho))$. Equations (13.44) and (13.45) imply that

$$\begin{aligned} e(\rho) &= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \left|\sigma_{[m]}[\nu]\, V^{2r}_{[m]}[\nu] - \hat{z}[\nu]_m\right|^2 \\ &= \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \left|\hat{z}[\nu]_m\right|^2 \left(\frac{\rho \left(2^{1-m}\ \sin(2^m \pi\, \nu/N)\right)^{2r}}{\rho \left(2^{1-m}\ \sin(2^m \pi\, \nu/N)\right)^{2r} + U^{2r}_{[m]}[\nu]}\right)^2. \end{aligned}$$

The derivative $e'(\rho) > 0 \; \forall \rho > 0$. Thus, the function $e(\rho)$ grows monotonically,

$$e(0) = 0, \quad \lim_{\rho \to \infty} e(\rho) = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \left| \hat{z}[\nu]_m \right|^2 = \sum_{l=0}^{N_m-1} z[k]^2. \tag{13.46}$$

Equation (13.46) means that, when $\rho = 0$, the spline $\mathbf{s}_{[m]}(\rho)$ interpolates the data vector $\mathbf{z}$. In the other limit case $\rho = \infty$, the spline $\mathbf{s}_{[m]}(\rho)$ interpolates the zero vector. Since the interpolating spline is unique, $\mathbf{s}_{[m]}(\infty) \equiv 0$. Thus, the approximation of the available data deteriorates while the regularity increases as ρ is growing and vice versa.

Applying the same reasoning as in the continuous case (Sect. 5.1.1), we come to the equation

$$e(\rho) = \frac{1}{N_m} \sum_{\nu=N_m/2}^{N_m/2-1} \left| \hat{z}[\nu]_m \right|^2 \left(\frac{\rho \left(2^{1-m} \, \sin(2^m \pi \nu / N)\right)^{2r}}{\rho \left(2^{1-m} \, \sin(2^m \pi \nu / N)\right)^{2r} + U_{[m]}^{2r}[\nu]} \right)^2 = N_m \, \varepsilon^2,$$

where $\varepsilon^2 = \mathrm{var}(\mathbf{e})$. The solution $\bar{\rho}$ of this equation, which exists and is unique, is an optimal value of the regularization parameter. The MATLAB function `D_defroP.m` derives the optimal regularization parameter $\bar{\rho}$. The values of smoothing discrete splines are computed by the MATLAB function `D_smoothsplaP.m`

Remark 13.1.6 The discrete spline $\mathbf{s}_{[m]}(\bar{\rho})$ provides an unique solution to the constrained minimization problem: Find a signal from $\mathbf{y} \in \Pi[N]$, which minimizes the functional $I(\mathbf{y}) \stackrel{\mathrm{def}}{=} \|\Delta^r[\mathbf{y}]\|^2$ subject to the condition that the discrepancy functional $E(\mathbf{y}) \stackrel{\mathrm{def}}{=} \sum_{l=0}^{N_m-1} (y[2^m \, l] - z[l])^2 \leq N_m \, \varepsilon^2$.

13.1.6.3 Examples

The following Fig. 13.5 displays results of experiments with discrete smoothing splines, which are similar to the experiments with continuous smoothing splines illustrated in Fig. 6.6. The experiments, which are implemented by the MATLAB code `D_smoothing_example_1_sin.m`, consist of approximation of the *chirp* curve depicted in Fig. 6.5 from the sparse noised data. In the first experiment, the data contains 128 equidistant samples of the function, which are affected by white noise whose STD=0.35. The sampled initial curve was approximated by the smoothing discrete splines $\mathbf{s}_{[3]}^{2r}$, which belong to $\Pi[2048]$. The splines orders are 4, 8 and 12, and the results are displayed in the top, middle and bottom of the left-hand side of Fig. 13.5, respectively. In the second experiment, the data was decimated by factor of 2, thus, the initial data consisted of 64 noised samples. The initial curve was approximated by the smoothing discrete splines $\mathbf{s}_{[4]}^{2r}$, which belong to $\Pi[2048]$. The splines orders are 4, 8 and 12, and the results are displayed in the top, middle and bottom of the right-hand side of Fig. 13.5, respectively.

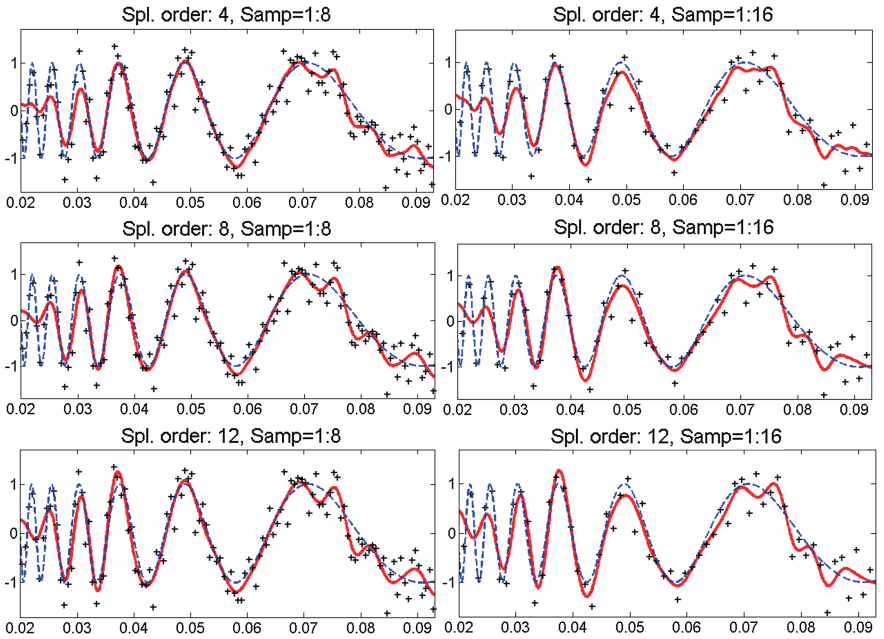


Fig. 13.5 *Left* restoration of the sampled function $f(t) = \sin(1/t)$ from 128 noised samples by the discrete smoothing splines of order 4 (*top*), 8 (*middle*) and 12 (*bottom*). *Right* restoration from 64 samples. *Dotted line*—original function, "pluses"—available data, *solid lines*—restoring splines

Comparing the above figures with Fig. 6.6, we observe that in the above examples the performance of discrete smoothing splines is very similar to the performance of continuous splines.

13.2 Two-dimensional Discrete Spline Harmonic Analysis

In this section, we expand the DSHA to the 2D case. This makes it applicable to processing digitally presented images.

13.2.1 Bases in Two-dimensional Discrete Spline Spaces

For simplicity, we assume that two-dimensional discrete splines to be discussed have the same periodicity N in both vertical and horizontal directions. The 2D N-periodic discrete B-spline of even order $p = 2r$ whose span is power of 2: $K = 2^m$ is defined as a product of 1D B-splines

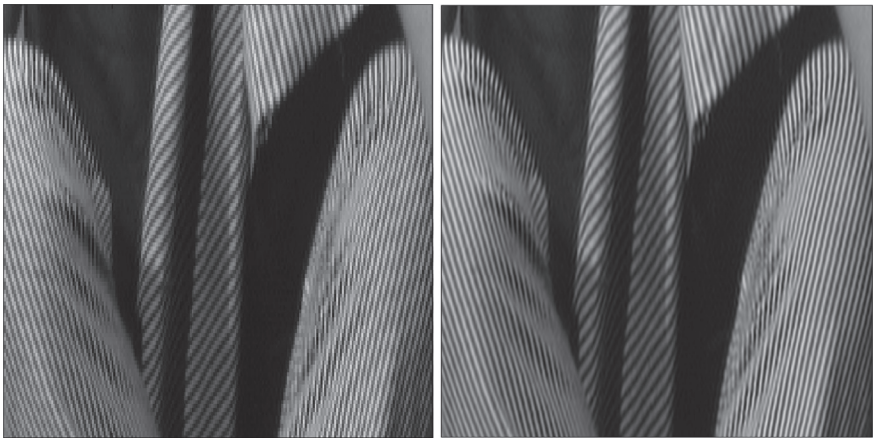


Fig. 13.6 *Left* a fragment of the original "Barbara" image. *Right* the same fragment of the image upsampled at the rate 1:8 using the discrete spline of eighth order

$$\mathbf{b}_{[m]}^{2r,2r} = \left\{ b_{[m]}^{2r}[k,n] \stackrel{\text{def}}{=} b_{[m]}^{2r}[k]\, b_{[m]}^{2r}[n] \right\}, \quad k,n \in \mathbb{Z}.$$

Similarly, the N-periodic 2D normalized exponential discrete splines

$$\begin{aligned} \gamma_{[m]}^{2r}(\kappa,\nu) &= \left\{ \gamma_{[m]}^{2r}(\kappa,\nu)[k,n] \stackrel{\text{def}}{=} \gamma_{[m]}^{2r}(\kappa)[k]\, \gamma_{[m]}^{2r}(\nu)[n] \right\}, \\ \kappa,\nu &= -N_m/2, \ldots, N_m/2-1, \quad N_m = N/2^m, \end{aligned}$$

are defined. 2D discrete splines are defined as linear combinations of the 2D basis splines

$$s_{[m]}^{2r}[k,n] = \sum_{l,\lambda=0}^{N_m-1} s[l,\lambda]\, b_{[m]}^{2r}[k-2^m l]\, b_{[m]}^{2r}[n-2^m\lambda]. \tag{13.47}$$

The space of such splines is denoted by ${}^{2r,2r}\mathscr{S}_{[m]}$. Alternatively,

$$s_{[m]}^{2r}[k,n] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa,\nu]\ \ \gamma_{[m]}^{2r}(\kappa,\nu)[k,n]. \tag{13.48}$$

Define the characteristic function of the space ${}^{2r,2r}\mathscr{S}_{[m]}$ as

$$U_{[m]}^{2r}[\kappa,\nu] \stackrel{\text{def}}{=} U_{[m]}^{2r}[\kappa]\, U_{[m]}^{2r}[\nu].$$

Coefficients of the expansions (13.47) and (13.48) of a spline $\mathbf{s}^{2r}_{[m]}$ are linked via the 2D DFT.

$$\sigma[\kappa,\nu] = \sqrt{U^{4r}_{[m]}[\kappa,\nu]} \sum_{l,\lambda=0}^{N_m-1} \omega^{-2^m(l\kappa+\lambda\nu)}\, s[l.\lambda,n],$$

$$s[l,\lambda] = \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \omega^{2^m(k\kappa+\lambda\nu)}\, \frac{\sigma[\kappa,\nu]}{\sqrt{U^{4r}_{[m]}[\kappa,\nu]}}.$$

The splines $\left\{\gamma^{2r}_{[m]}(\kappa,\nu)\right\}.\ \kappa,\nu = -N_m/2,\ldots,N_m/2-1$, form an orthonormal basis of ${}^{2r,2r}\mathscr{S}_{[m]}$.

13.2.2 Outline of 2D DSHA Relations

Extension of the DSHA to 2D spline spaces is straightforward. In particular,

Sampling and interpolation: Let $\mathbf{s}^{2r}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$ be represented as in Eq. (13.48). Then, Eq. (13.23) implies that grid samples

$$\begin{aligned} s^{2r}_{[m]}[2^m l, 2^m\lambda] &= \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa,\nu]\, \gamma^{2r}_{[m]}(\kappa,\nu)[2^m l, 2^m\lambda] \qquad (13.49)\\ &= \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa,\nu]\, V^{2r}_{[m]}[\kappa]\, V^{2r}_{[m]}[\nu]\, \omega^{2^m(\kappa l+\nu\lambda)},\\ V^{2r}_{[m]}[\nu] &\stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}}. \end{aligned}$$

If the spline $\mathbf{s}^{2r}_{[m]}$ interpolates an array $\mathbf{z} = \{z[l,\lambda]\}\,,\ l,\lambda,=0,\ldots,N_m-1$, then

$$\begin{aligned} s^{2r}_{[m]}[2^m l, 2^m\lambda] &= z[l,\lambda] \Longleftrightarrow \sigma[\kappa,\nu] = \frac{\hat{z}[\kappa,\nu]_m}{V^{2r}_{[m]}[\kappa]\, V^{2r}_{[m]}[\nu]}, \qquad (13.50)\\ \hat{z}[\kappa,\nu]_m &= \sum_{l,\lambda,=0}^{N_m-1} \omega^{2^m(l\kappa+\lambda\nu)}\, z[l,\lambda]. \end{aligned}$$

Finite differences: The central finite difference $\delta^{2d,2\bar{d}}[\mathbf{x}]$ for 2D periodic array $\mathbf{x}$ was defined in Eq. (1.34). In particular, $\delta^{2d,0}[\mathbf{x}]$ and $\delta^{0,2\bar{d}}[\mathbf{x}]$ are the partial differences of $\mathbf{x}$ in vertical and horizontal directions, respectively. Let $\mathbf{s}^{2r}_{[m]} \in$

${}^{2r,2r}\mathscr{S}_{[m]}$ be represented as in Eq. (13.48). Then, similarly to Eq. (13.25) we derive, for $d, \bar{d} < r$, using Eq. (1.35):

$$\begin{aligned}
\delta^{2d,2\bar{d}}[\mathbf{s}^{2r}_{[m]}][k,n] &= \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} B^{r,d,\bar{d}}_{[m]}[\kappa,\nu]\, \sigma_{[m]}[\kappa,\nu]\, \gamma^{2(r-d)}_{[m]}(\kappa)[k]\, \gamma^{2(r-\bar{d})}_{[m]}(\nu)[n], \\
B^{r,d,\bar{d}}_{[m]}[\kappa,\nu] &\stackrel{\text{def}}{=} B^{r,d}_{[m]}[\kappa]\, B^{r,\bar{d}}_{[m]}[\nu], \\
B^{r,d}_{[m]}[\nu] &\stackrel{\text{def}}{=} \sqrt{\frac{U^{4(r-d)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}} \left(\frac{2\sin(2^m \pi\, \nu/N)}{2^m}\right)^{2d}.
\end{aligned} \tag{13.51}$$

Parseval identities: Let $\mathbf{s}^{2r}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$ be represented as in Eq. (13.48). Similarly to Eq. (13.29), we obtain

$$\left\| \delta^{2d,2\bar{d}}[\mathbf{s}_{[m]}] \right\|^2 = \frac{1}{N_m^2} \sum_{\kappa,\nu=N_m/2}^{N_m/2-1} |\sigma_{[m]}[\kappa,\nu]|^2 \left(B^{r,d,\bar{d}}_{[m]}[\kappa,\nu] \right)^2, \tag{13.52}$$

where the sequence $B^{r,d,\bar{d}}_{[m]}[\kappa,\nu]$ is defined in Eq. (13.51).

Circular convolution: Let $\mathbf{s}^{2r}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$ be represented as in Eq. (13.48) and

$$\bar{s}^{2q}_{[m]}[k,n] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \bar{\sigma}[\kappa,\nu]\ \ \gamma^{2q}_{[m]}(\kappa,\nu)[k,n] \in {}^{2q,2q}\mathscr{S}_{[m]}.$$

Then the discrete circular convolution, which is a discrete spline from ${}^{2(r+q),2(r+q)}\mathscr{S}_{[m]}$

$$\begin{aligned}
\mathbf{s}_{[m]} \circledast \bar{s}_{[m]}[k,n] &= \sum_{l\lambda=0}^{N-1} \mathbf{s}_{[m]}[k-l, n-\lambda]\, \bar{s}_{[m]}[l,\lambda] \\
&= \frac{1}{N_m} \sum_{\kappa,\nu=N_m/2}^{N_m/2-1} \sigma_{[m]}[\kappa,\nu]\, \bar{\sigma}_{[m]}[\kappa,\nu]\, \bar{R}^{r,q}_{[m]}(\kappa,\nu)\, \gamma^{2(r+q)}_{[m]}(\kappa,\nu)[k,n],
\end{aligned}$$

where $R^{p,q}[\kappa,\nu] \stackrel{\text{def}}{=} \sqrt{\dfrac{U^{4(r+q)}_{[m]}[\kappa,\nu]}{U^{4r}_{[m]}[\kappa,\nu]\, U^{4q}_{[m]}[\kappa,\nu]}}$.

13.2.3 Explicit Computation of a 2D Periodic Discrete Spline

Similarly to 1D periodic discrete splines, values of 2D discrete splines can be explicitly calculated using the 2D DFT. Assume that a spline $\mathbf{s}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$ is expanded over two alternative bases

$$
\begin{aligned}
s^{2r}_{[m]}[k,n] &= \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa,\nu]\ \ \gamma^{2r}_{[m]}(\kappa,\nu)[k,n] \\
&= \sum_{l,\lambda=0}^{N_m-1} q[l,\lambda]\, b^{2r}_{[m]}[k-2^m l]\, b^{2r}_{[m]}[n-2^m\lambda], \\
\hat{q}[\kappa,\nu]_m &= \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \omega^{-2^m(l\kappa+\lambda\nu)}\, s[l,\lambda] = \frac{\sigma[\kappa,\nu]}{\sqrt{U^{4r}_{[m]}[\kappa,\nu]}}.
\end{aligned}
$$

Then, the 2D DFT of the spline $\mathbf{s}_{[m]}$ is

$$
\begin{aligned}
\hat{\mathbf{s}}_{[m]^{2r}}[\kappa,\nu] &= \sum_{k,n=0}^{N-1} \omega^{-(k\kappa+n\nu)}\, s^{2r}_{[m]}[k,n] = \hat{q}[\kappa,\nu]_m\, \hat{b}^{2r}_{[m]}[\kappa,\nu] \qquad (13.53) \\
&= \frac{\sigma[\kappa,\nu]}{\sqrt{U^{4r}_{[m]}[\kappa,\nu]}}\, \Omega^{2r}[\kappa,\nu], \\
\Omega[\kappa,\nu] &\stackrel{\text{def}}{=} \frac{\sin(2^m\pi\,\kappa/N)}{2^m\,\sin(\pi\,\kappa/N)}\, \frac{\sin(2^m\pi\,\nu/N)}{2^m\,\sin(\pi\,\nu/N)}.
\end{aligned}
$$

Thus, the spline $\mathbf{s}_{[m]}$ is calculated via the 2D IDFT:

$$
s^{2r}_{[m]}[k,n] = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \frac{\omega^{(k\kappa+n\nu)}\,\sigma[\kappa,\nu]}{\sqrt{U^{4r}_{[m]}[\kappa,\nu]}}\ \ \Omega^{2r}[\kappa,\nu]. \qquad (13.54)
$$

In particular, if grid samples of the spline $\mathbf{s}_{[m]}$ are $s^{2r}_{[m]}[2^m l, 2^m\lambda] = x[l,\lambda]$ then, due to Eq. (13.50),

$$
\begin{aligned}
\sigma[\kappa,\nu] &= \frac{\hat{x}[\kappa,\nu]_m}{V^{2r}_{[m]}[\kappa]\, V^{2r}_{[m]}[\nu]}, \quad V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}} \qquad (13.55) \\
\Longrightarrow s^{2r}_{[m]}[k,n] &= \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \frac{\omega^{(k\kappa+n\nu)}\,\hat{x}[\kappa,\nu]_n}{U^{2r}_{[m]}[\kappa,\nu]}\, \Omega^{2r}[\kappa,\nu].
\end{aligned}
$$

These calculations are implemented by the MATLAB function `D_interspll_2dP.m`.

Remark 13.2.7 Equation (13.55) provide an efficient tool for upsampling digitally presented images. For this, in order to upsample the initial array of pixels $\mathbf{x}$ at the rate $1:2^m$, it should be interpolated by a discrete spline $\mathbf{s}_{[m]}$, whose values are calculated by Eq. (13.55).

Figure 13.6, which is produced by the MATLAB code `D_upsam_exam.m`, illustrates upsampling of a fragment of the "Barbara" image, which originally was presented by an array of 512×512 pixels. A fragment of the original image is displayed on the left frame, whereas the right frame displays the respective fragment of the upsampled image at the rate 1:8 image. The upsampled image demonstrates much better resolution of the stripes.

13.2.4 2D Smoothing Periodic Discrete Splines

2D smoothing periodic discrete splines are the direct extension of 1D smoothing splines. The design scheme is the same as in 1D case. Assume that samples $x[2^m l, 2^m \lambda],\ l, \lambda = 0, \ldots, N_m$, of an $N-$periodic in both directions array $\mathbf{x} = \{x[k, n]\} \in \Pi[N, N]$ are corrupted by zero mean Gaussian noise $\mathbf{e} = \left\{e_{l,\lambda}\right\},\ l, \lambda = 0, \ldots, N_m$, whose variance $\mathrm{var}(\mathbf{e}) = \varepsilon^2$. Thus, we approximate the array $\mathbf{x} \in \Pi[N, N]$ by an array $\mathbf{y} \in \Pi[N, N]$ derived from the sparse corrupted data $\mathbf{z} = \{z[l, \lambda] = x[2^m l, 2^m \lambda] + e_l\},\ l, \lambda = 0, \ldots, N_m$.

13.2.4.1 Parameterized Discrete Splines

Assume that $r = 2d$ is an even number. We look for a spline

$$\mathbf{s}^{2r}(\rho) = \left\{s^{2r}(\rho)[k, n]\right\} \in {}^{2r,2r}\mathscr{S}_{[m]} \subset \Pi[N, N],$$

which minimizes the parameterized functional $J_\rho(\mathbf{s}) \stackrel{\mathrm{def}}{=} \rho I(\mathbf{s}) + E(\mathbf{s})$, where

$$I(\mathbf{s}) \stackrel{\mathrm{def}}{=} \left\|\delta_k^r[\mathbf{s}]\right\|^2 + \left\|\delta_n^r[\mathbf{s}]\right\|^2, \quad E(\mathbf{s}) \stackrel{\mathrm{def}}{=} \sum_{l,\lambda=0}^{N_m-1} (s[2^m l, 2^m \lambda] - z[l, \lambda])^2.$$

When $\rho = 0$, the minimum of $J_\rho(\mathbf{s})$ is achieved on the discrete spline $\mathbf{s}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$, which interpolates the data $\mathbf{z}$. This spline is presented in Eq. (13.55).

The spline $\mathbf{s}_{[m]}(\rho) \in {}^{2r,2r}\mathscr{S}_{[m]}$, which minimizes the functional J_ρ, is called the 2D discrete periodic smoothing spline.

Assume that a spline $\mathbf{s}_{[m]} \in {}^{2r,2r}\mathscr{S}_{[m]}$ is represented via the orthonormal basis as in Eq. (13.48)

$$s_{[m]}[k, n] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa, \nu]\ \ \gamma_{[m]}^{2r}(\kappa, \nu)[k, n].$$

Then, Eq. (13.52) implies that

$$I(\mathbf{s}_{[m]}) \;=\; \left\| \delta^{2d}[\mathbf{s}_{[m]}] \right\|^2 = \frac{1}{N_m^2} \sum_{\kappa,\nu=N_m/2}^{N_m/2-1} |\sigma_{[m]}[\kappa,\nu]|^2 \;\left(W_{[m]}^{r,r}(\kappa)] + W_{[m]}^{r,r}[\nu] \right),$$

where $W_{[m]}^{r,r}[\nu] \;\stackrel{\text{def}}{=}\; \left(\dfrac{2\sin(2^m \pi\, \nu/N)}{2^m} \right)^{2r} \dfrac{U_{[m]}^{2r}[\nu]}{U_{[m]}^{4r}[\nu]}.$

Equation (13.49) provides grid samples of the spline

$$s_{[m]}[2^m l, 2^m \lambda] \;=\; \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma[\kappa,\nu]\, V_{[m]}^{2r}[\kappa]\, V_{[m]}^{2r}[\nu]\, \omega^{2^m(\kappa l+\nu\lambda)},$$

$$V_{[m]}^{2r}[\nu] \;\stackrel{\text{def}}{=}\; \frac{U_{[m]}^{2r}[\nu]}{\sqrt{U_{[m]}^{4r}[\nu]}},$$

which are represented via the 2D IDFT. Similarly, the data array is represented as

$$z[l,\lambda] = \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \omega^{2^m(\kappa l+\nu\lambda)}\, \hat{z}[\kappa,\nu]_m.$$

Then, using the Parseval identity Eq. (1.33), we have

$$\begin{aligned} E(\mathbf{s}_{[m]}) \;&\stackrel{\text{def}}{=}\; \sum_{l,\lambda=0}^{N_m-1} (s_{[m]}[2^m\, l, 2^m\, \lambda] - z[l,\lambda])^2 \qquad (13.56) \\ &= \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \left| \sigma_{[m]}[\kappa,\nu]\, V_{[m]}^{2r}[\kappa]\, V_{[m]}^{2r}[\nu] - \hat{z}[\kappa,\nu]_m \right|^2. \end{aligned}$$

Hence, we obtain an explicit expression of the parameterized functional

$$\begin{aligned} J_\rho(\mathbf{s}_{[m]}) = \frac{1}{N_m^2} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \Big(&\rho\, |\sigma_{[m]}[\kappa,\nu]|^2 \;\left(W_{[m]}^{r,r}[\kappa] + W_{[m]}^{r,r}[\nu] \right)] \Big) \\ &+ \left| \sigma_{[m]}[\kappa,\nu]\, V_{[m]}^{2r}[\kappa]\, V_{[m]}^{2r}[\nu] - \hat{z}[\kappa,\nu]_m \right|^2 \Big). \end{aligned}$$

Proposition 13.11 *Minimum to the functional* $J_\rho(\mathbf{s}_{[m]})$ *on the discrete spline space* ${}^{2r,2r}\mathcal{S}_{[m]}$ *is provided by the spline*

$$
\begin{aligned}
s_{[m]}(\rho)[k,n] &= \frac{1}{N_m}\sum_{\kappa,\nu=N_m/2}^{N_m/2-1}\sigma_{[m]}[\kappa,\nu]\,\gamma^{2r}_{[m]}(\kappa,\nu)[k,n],\\
\sigma_{[m]}[\kappa,\nu] &= \frac{V^{2r}_{[m]}[\kappa]\,V^{2r}_{[m]}[\nu]\,\hat{z}[\kappa,\nu]_m}{\rho\left(W^{r,r}_{[m]}(\kappa)]+W^{r,r}_{[m]}[\nu]\right)+(V^{2r}_{[m]}[\kappa]\,V^{2r}_{[m]}[\nu])^2}\\
&= \frac{\sqrt{U^{4r}_{[m]}[\kappa,\nu]}\,\hat{z}[\nu]_m}{\bar{A}^r_\rho[\kappa,\nu]},\\
\bar{A}^r_\rho[\kappa,\nu] &\stackrel{\text{def}}{=} \rho\,2^{1-m}\left(\left(\sin\frac{2^m\pi\kappa}{N}\right)^{2r}+\left(\sin\frac{2^m\pi\nu}{N}\right)^{2r}\right)+U^{2r}_{[m]}[\kappa,\nu].
\end{aligned}
$$

The proof of this statement is similar to the proof of Proposition 5.3.

Using Eq. (13.54), we derive the explicit expression of the smoothing spline:

$$
s^{2r}_{[m]}(\rho)[k,n]=\frac{1}{N^2}\sum_{\kappa,\nu=-N/2}^{N/2-1}\omega^{(k\kappa+n\nu)}\;\frac{\hat{z}[\kappa,\nu]_m}{\bar{A}^r_\rho[\kappa,\nu]}\;\Omega^{2r}[\kappa,\nu],
$$

where the sequence $\Omega^{2r}[\kappa,\nu]$ is defined in Eq. (13.53). The grid samples of the spline $\mathbf{s}^{2r}_{[m]}(\rho)$ are

$$
\begin{aligned}
s_{[m]}(\rho)[2^m l,2^m\lambda] &= \frac{1}{N_m^2}\sum_{\kappa,\nu=N_m/2}^{N_m/2-1}\omega^{2^m(\kappa l+\nu\lambda)}\,\sigma_{[m]}[k,\nu]\,V^{2r}_{[m]}[\kappa]]\,V^{2r}_{[m]}[\nu]\\
&= \frac{1}{N_m^2}\sum_{\kappa,\nu=N_m/2}^{N_m/2-1}\omega^{2^m(\kappa l+\nu\lambda)}\,\frac{U^{2r}_{[m]}[\kappa,\nu]\,\hat{z}[\kappa,\nu]_m}{\bar{A}^r_\rho[\kappa,\nu]}
\end{aligned}
$$

Obviously, when $\rho=0$, the smoothing spline $\mathbf{s}_{[m]}(\rho)$ turns into an interpolatory one.

13.2.4.2 Optimal Regularization Parameter

Denote $e(\rho)\stackrel{\text{def}}{=}E(\mathbf{s}_{[m]}(\rho))$. Equations (13.56) and (13.45) imply that

$$
\begin{aligned}
e(\rho) &= \frac{1}{N_m^2}\sum_{\kappa,\nu=N_m/2}^{N_m/2-1}\left|\sigma_{[m]}[\kappa,\nu]\,V^{2r}_{[m]}[\kappa]\,V^{2r}_{[m]}[\nu]-\hat{z}[\kappa,\nu]_m\right|^2\\
&= \frac{1}{N_m^2}\sum_{\kappa,\nu=N_m/2}^{N_m/2-1}\left|\hat{z}[\kappa,\nu]_m\right|^2\left(\frac{\rho\,G^{2r}_{[m]}[\kappa,\nu]}{\rho\,G^{2r}_{[m]}[\kappa,\nu]+U^{2r}_{[m]}[\kappa,\nu]}\right)^2,
\end{aligned}
$$

Fig. 13.7 *Top left*—"Barbara" image corrupted by noise whose STD $\varepsilon = 15$, PSNR $= 24.62$ dB. *Top right*—cubic discrete smoothing spline, PSNR $= 26.52$ dB. *Bottom left*—the image corrupted by noise, $\varepsilon = 60$, PSNR $= 12.58$ dB. *Bottom right*—cubic discrete smoothing spline, PSNR $= 22.69$ dB

$$G^{2r}_{[m]}[\kappa, \nu] \stackrel{\text{def}}{=} \left(2^{1-m} \sin(2^m \pi \kappa / N)\right)^{2r} + \left(2^{1-m} \sin(2^m \pi \nu / N)\right)^{2r}.$$

As in the 1D case, we find an optimal value of ρ from the equation

$$e(\rho) = \frac{1}{N_m^2} \sum_{\kappa,\nu=N_m/2}^{N_m/2-1} \left|\hat{z}[\kappa, \nu]_m\right|^2 \left(\frac{\rho\, G^{2r}_{[m]}[\kappa, \nu]}{\rho\, G^{2r}_{[m]}[\kappa, \nu] + U^{2r}_{[m]}[\kappa, \nu]}\right)^2 = N_m^2\, \varepsilon^2, \tag{13.57}$$

where $\varepsilon^2 = \mathrm{var}(\mathbf{e})$. The solution $\bar{\rho}$ of Eq. (13.57), which exists and is unique, provides the optimal value of the regularization parameter.

Optimal parameter value is determined by the MATLAB function `D_defroP.m`, while the 2D discrete smoothing splines are designed by the MATLAB function `D_smoothsplaP_2d.m`.

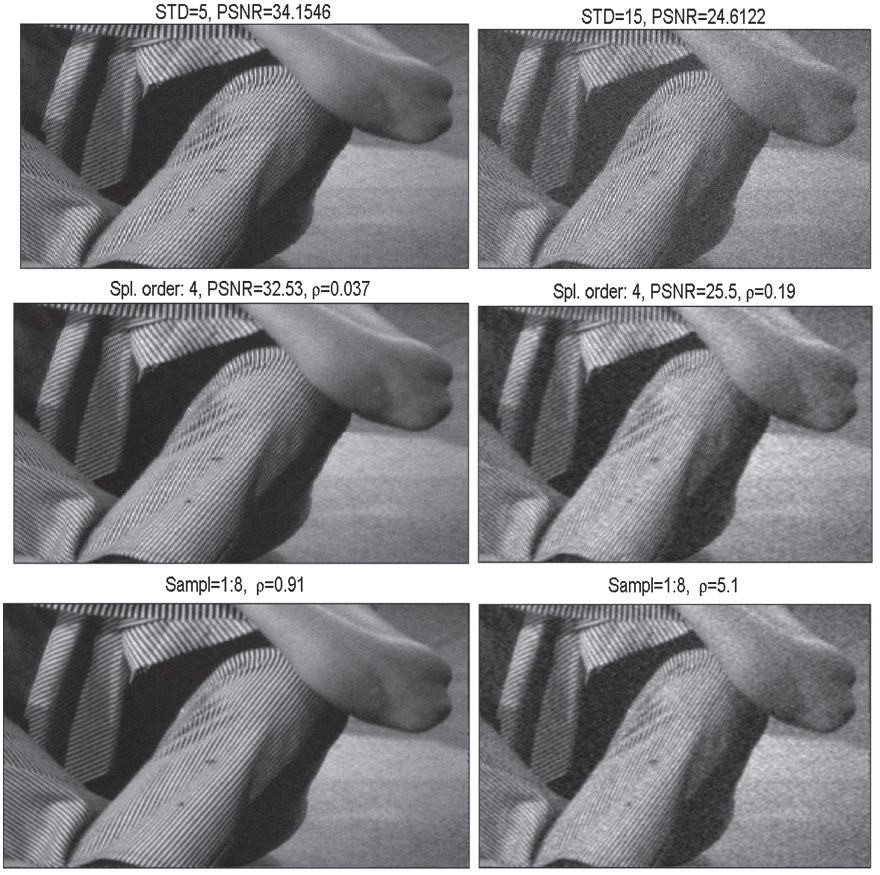


Fig. 13.8 *Top left*—a fragment of the "Barbara" image corrupted by noise whose STD $\varepsilon = 5$, PSNR $= 34.15$ dB. *Center left* cubic discrete smoothing spline, PSNR $= 32.53$ dB. *Bottom left* upsampled at the rate of 1:8 discrete cubic smoothing spline. *Top right*—a fragment of the "Barbara" image corrupted by noise whose STD $\varepsilon = 15$, PSNR $= 24.61$ dB. *Center right* cubic discrete smoothing spline, PSNR $= 25.5$ dB. *Bottom right* upsampled at the rate of 1:8 discrete cubic smoothing spline

13.2.4.3 Examples

A few numerical examples illustrate performance of the discrete smoothing splineson restoration of digital images from noised input data.

Restoration of the "Barbara" image from severely noised non-decimated data

Figure 13.7, which is produced by the MATLAB code `D_smooth_example_opt_2dP.m`, illustrates results from the denoising experiment where the "Barbara" image was restored from non-decimated noised input arrays. The top frame demonstrates restoration of the image from data, which was corrupted by a relatively moderate noise, whose STD $\varepsilon = 15$ by the discrete smooth-

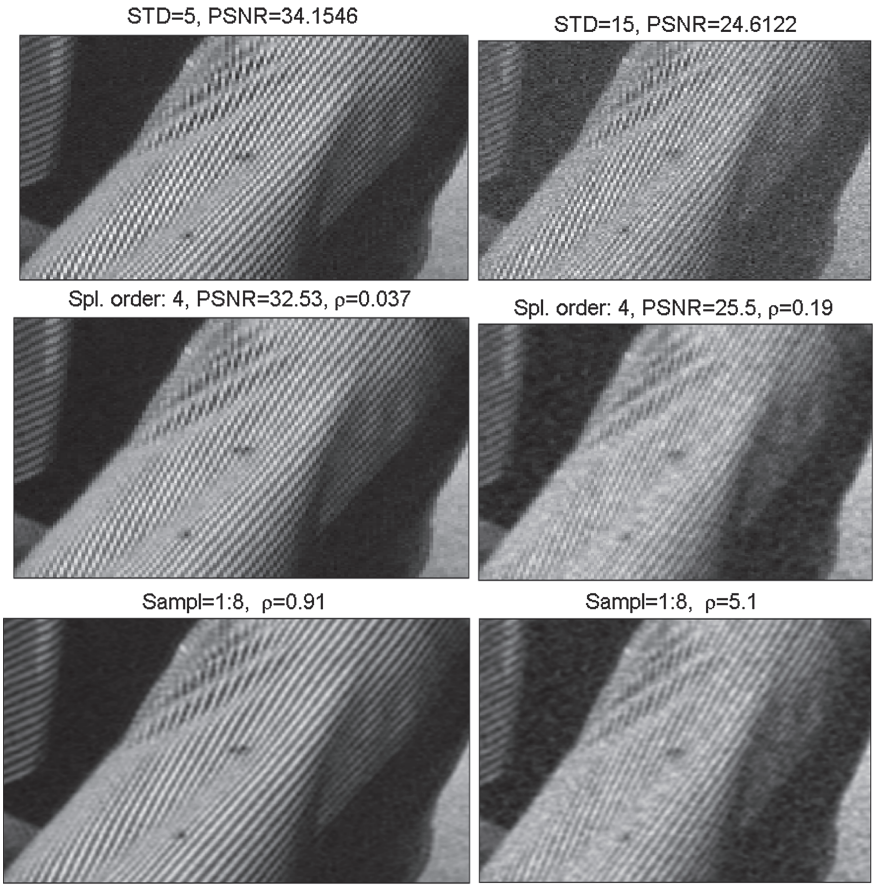


Fig. 13.9 Closed-up fragments of images from Fig. 13.8

ing spline of fourth order. The noise is significantly reduced while the structure of the image is retained. The PSNR is elevated from 24.62 dB to 26.52 dB. The value of the parameter ρ derived from Eq. (13.57) is $\bar{\rho} = 0.31$.

The bottom frame demonstrates restoration of the image from severely corrupted data – noise STD $\varepsilon = 60$. PSNR of the corrupted image is 12.58 dB. The image restored by application of the fourth order discrete smoothing splines has PSNR 22.69 dB. The main structure of the image, which was buried in noise, was extracted. Because the relative contribution of noise to the available data is high, the automatically determined value of $\bar{\rho}$ is much bigger than in the former case: $\bar{\rho} = 11$. This results in significant suppression of noise. However, the fine texture in the image became oversmoothed.

Restoration and upsampling the "Barbara" image from noised data
Figure 13.8, which was produced by the MATLAB code
`D_smooth_upsam_exam_2dP.m`, illustrates results from the experiment, where denoising was coupled with upsampling the "Barbara" image. The left-hand triple of frames from top to bottom display:

1. A fragment of image corrupted by a relatively weak noise, whose STD $\varepsilon = 5$.
2. The fragment restored by the discrete smoothing spline of fourth order.
3. The fragment restored by the upsampled at the rate of 1:8 discrete smoothing spline of fourth order.

The right-hand triple of frames display:

1. The fragment of image corrupted by noise, whose STD $\varepsilon = 15$.
2. The fragment restored by the discrete smoothing spline of fourth order.
3. The fragment restored by the upsampled at the rate of 1:8 discrete smoothing spline of fourth order.

One can observe that the upsampled splines produce better denoising and definitely overcome the non-upsampled splines on resolution of stripes in the image. This is clearly seen in Fig. 13.9, which displays closed up fragments of previous images.

Note that when noise was weak, application of the non-upsampled smoothing spline (center left frame in Fig. 13.8) results in reduction of the PSNR.

References

1. V.N. Malozemov, S.M. Masharsky, *Basics of the Discrete Harmonic Analysis* (Lan', St. Petersburg, 2012) (In Russian)
2. V.N. Malozemov, A.B. Pevnyi, Discrete periodic splines and their numerical application. Comp. Math. Math. Phys **38**, 1181–1192 (1998)
3. E.T. Whittaker, On a new method of graduation. Proc. Edinburgh Math. Soc. **41**, 63–75 (1922)

Chapter 14
Discrete Periodic Spline Wavelets and Wavelet Packets

Abstract Similarly to periodic polynomial splines, existence of the set of embedded discrete periodic splines spaces $\Pi[N] = \mathcal{S}_{[0]} \supset {}^{2r}\mathcal{S}_{[1]} \supset \cdots \supset {}^{2r}\mathcal{S}_{[m]} \cdots$, combined with the DSHA provides flexible tools for design and implementation of wavelet and wavelet packet transforms. As in the polynomial case, all the calculations consist of fast direct and inverse Fourier transforms (FFT and IFFT, respectively) and simple arithmetic operations. Raising the splines order does not increase the computation complexity.

In this chapter, sets of mutually In this chapter, sets of mutually orthogonal subspaces of the periodic signals space are designed. The subspaces are related to different frequency bands. They consist of discrete splines and discrete spline wavelets and wavelet packets to be designed. The design is based on two-scale relations between the discrete splines from different spaces ${}^{2r}\mathcal{S}_{[m]}$.

14.1 Two-Scale Relations

Proposition 13.2 establishes the following relation between discrete exponential splines of different span

$$\zeta^{2r}_{[m]}(\nu)[k] = \frac{1}{2}\left(b_{[m-1]}(\nu)\,\zeta^{2r}_{[m-1]}(\nu)[k] + b_{[m-1]}(\nu+N_m)\,\zeta^{2r}_{[m-1]}(\nu+N_m)[k]\right),$$
$$b_{[m-1]}(\nu) \stackrel{\text{def}}{=} \cos^{2r}\frac{\pi\,\nu}{N_{m-1}}. \tag{14.1}$$

Equation 14.1 leads to the relation between the normalized exponential discrete splines $\gamma^{2r}_{[m]}(\nu)[k] = \zeta^{2r}_{[m]}(\nu)[k]/\sqrt{N_m\,U^{4r}_{[m]}[\nu]}$:

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_14,

$$\gamma_{[m]}^{2r}(\nu)[k] = s_{[m-1]}(\nu)\ \gamma_{[m-1]}^{2r}(\nu)[k] + s_{[m-1]}(\nu+N_m)\ \gamma_{[m-1]}^{2r}(\nu+N_m)[k],$$

$$s_{[m-1]}(\nu) \stackrel{\text{def}}{=} \sqrt{\frac{U_{[m-1]}^{4r}[\nu]}{2\,U_{[m]}^{4r}[\nu]}}\,\cos^{2r}\frac{\pi\,\nu}{N_{m-1}}. \tag{14.2}$$

The norms of the splines $\gamma_{[m]}^{2r}(\nu)$ and $\gamma_{[m-1]}^{2r}(\nu)$ are equal to one. Hence, we have

$$(s_{m-1}(\nu))^2 + (s_{m-1}(\nu+N_m))^2 = 1, \quad \nu = -N_m/2, \ldots, N_m/2-1. \tag{14.3}$$

14.1.1 Orthogonal Complements of Discrete Spline Spaces

Denote, by ${}^{2r}\mathscr{W}_{[m],1}$ the orthogonal complement to the spline space ${}^{2r}\mathscr{S}_{[m]}$ in the space ${}^{2r}\mathscr{S}_{[m-1]}$. We construct an orthonormal basis that characterizes N_m-dimensional space ${}^{2r}\mathscr{W}_{[m],1}$. Define the splines for $\nu = -N_m/2, ..., N_m/2-1$

$$\gamma_{[m],1}^{2r}(\nu)[k] \stackrel{\text{def}}{=} d_{m-1}(\nu)\,\gamma_{[m-1]}^{2r}(\nu)[k] + d_{m-1}(\nu+N_m)\,\gamma_{[m-1]}^{2r}(\nu+N_m)[k], \tag{14.4}$$

$$d_{m-1}(\nu) \stackrel{\text{def}}{=} \omega^{2^{m-1}\nu}\,s_{m-1}(\nu+N_m) = \omega^{2^{m-1}\nu}\sqrt{\frac{U_{[m-1]}^{4r}[\nu+N_m]}{2\,U_{[m]}^{4r}[\nu]}}\,\sin^{2r}\frac{\pi\,\nu}{N_{m-1}}.$$

There holds a statement similar to Proposition 8.3.

Proposition 14.1 *The set of discrete splines* $\{\gamma_{[[m],1}^{2r}(\nu)\}$, $\nu = -N_m/2, ..., N_m/2-1$, *forms an orthonormal basis for* ${}^{2r}\mathscr{W}_{[m],1}$.

Proof Due to Eq. (14.3), the squared norms

$$\begin{aligned}\left\|\gamma_{[m],1}^{2r}(\nu)\right\|^2 &= |d_{m-1}(\nu)|^2\left\|\gamma_{[m-1]}^{2r}[\nu]\right\|^2 + |d_{m-1}(\nu+N_m)|^2\left\|\gamma_{[m-1]}^{2r}(\nu+N_m)\right\|^2\\ &= (s_{m-1}(\nu+N_m))^2 + (s_{m-1}(\nu))^2 = 1.\end{aligned}$$

The proof of mutual orthogonality of the splines $\{\gamma_{[m],1}^{2r}(\nu)\}$, $\nu = -N_m/2, ..., N_m/2-1$, and the orthogonality relation between the splines $\gamma_{[m]}^{2r}(\nu) \in {}^{2r}\mathscr{S}_{[m]}$ and $\gamma_{[m],1}^{2r}(\nu) \in {}^{2r}\mathscr{W}_{[m],1}$ is identical to the proof of the respective pieces in Proposition 8.3. ■

Proposition 14.2 *The discrete splines* $\gamma_{[m]}^{2r}(\nu)$ *and* $\gamma_{[m],1}^{2r}(\nu)$ *are the eigenvectors of the shift operator*

$$\gamma_{[m]}^{2r}(\nu)[k+2^m l] = \omega^{2^m\nu l}\,\gamma_{[m]}^{2r}(\nu)[k], \tag{14.5}$$

$$\gamma_{[m],1}^{2r}(\nu)[k+2^m l] = \omega^{2^m\nu l}\,\gamma_{[m],1}^{2r}(\nu)[k], \quad l = 0, \ldots, N_m-1. \tag{14.6}$$

Proof At the initial space $\mathcal{S}_{[0]}$, the splines $\gamma_{[0]}(\nu)[k] = \omega^{\nu k}/\sqrt{N}$. Thus

$$\gamma_{[0]}(\nu)[k+l] = \frac{1}{\sqrt{N}}\,\omega^{\nu(k+l)} = \omega^{\nu l}\,\gamma^{2r}_{[0]}(\nu)[k].$$

Relations Eqs. (14.5) and (14.6) are derived by induction from the two-scale relations Eqs. (14.2) and (14.4). ■

The union $\left\{\gamma^{2r}_{[m]}(\nu)\right\} \bigcup \left\{\gamma^{2r}_{[m],1}(\nu)\right\}$, $\nu = -N_m/2, ..., N_m/2-1$, forms an orthonormal basis for the space ${}^{2r}\mathcal{S}_{[m-1]}$, while the union

$$\left\{\gamma^{2r}_{[m]}(\nu)\right\} \bigcup \left\{\gamma^{2r}_{[m],1}(\nu)\right\} \bigcup \left\{\gamma^{2r}_{[m-1],1}(\nu)\right\} ... \bigcup \left\{\gamma^{2r}_{[1],1}(\nu)\right\}$$

forms an orthonormal basis for the entire signal space $\Pi[N]$.

14.1.2 Refined Split of the Signal Space $\Pi[N]$ into Orthogonal Subspaces

The split scheme is the same as the scheme of the split of polynomial splines spaces as it is described in Sect. 8.2.3. For uniformity, we re-denote the spline spaces ${}^{2r}\mathcal{W}_{[m],0} \stackrel{\text{def}}{=} {}^{2r}\mathcal{S}_{[m]}$, $\mathcal{W}_{[0]} \stackrel{\text{def}}{=} \mathcal{S}_{[0]} = \Pi[N]$ and the exponential splines $\gamma^{2r}_{[m],0}(\nu) \stackrel{\text{def}}{=} \gamma^{2r}_{[m]}(\nu)$.

The zero level normalized exponential splines are simply the normalized Fourier exponentials $\left\{\gamma_{[0]}(\nu)[k] = \omega^{\nu k}/\sqrt{N}\right\}$, $\nu = -N/2, ..., N/2-1$. Thus, representation of a signal $\mathbf{x} = \{x[k]\} \in \Pi[N] = \mathcal{S}_{[0]}$ via its DFT

$$x[k] = \frac{1}{N}\sum_{\nu=-N/2}^{N/2-1} \hat{x}[\nu]\,\omega^{\nu k} = \frac{1}{\sqrt{N}}\sum_{\nu=-N/2}^{N/2-1} \hat{x}[\nu]\,\gamma_{[0]}(\nu)[k], \text{ where } (14.7)$$

$$\hat{x}[\nu] \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} x[k]\,\omega^{-\nu k} = \sqrt{N}\,\left\langle \mathbf{x},\ \gamma_{[0]}(\nu)\right\rangle,$$

can be regarded as its expansion over the orthonormal system $\Gamma_{[0]} \stackrel{\text{def}}{=} \left\{\gamma_{[0]}(\nu)\right\}$, $\nu = -N/2, ..., N/2-1$. In that sense, the space $\mathcal{S}_{[0]}$ is a linear span of the set $\Gamma_{[0]}$.

The space $\mathcal{S}_{[0]}$ is split into the mutually orthogonal subspaces of the first decomposition level $\mathcal{S}_{[0]} = {}^{2r}\mathcal{W}_{[1],0} \bigoplus {}^{2r}\mathcal{W}_{[1],1}$. The subspaces ${}^{2r}\mathcal{W}_{[1],0}$ and ${}^{2r}\mathcal{W}_{[1],1}$ are the linear spans of the orthonormal sets $\Gamma_{[1],0} \stackrel{\text{def}}{=} \left\{\gamma^{2r}_{[1],0}(\nu)\right\}$ and $\Gamma_{[1],1} \stackrel{\text{def}}{=} \left\{\gamma_{[1],1}(\nu)\right\}$, respectively, where $\nu = -N_1/2, ..., N_1/2-1$. This mutually orthogonal pair of orthonormal sets $\Gamma_{[1]} \stackrel{\text{def}}{=} \Gamma_{[1],0} \bigoplus \Gamma_{[1],1}$, which constitutes an alternative

orthonormal basis for $\Pi[N]$, is derived from the system $\Gamma_{[0]}$ by the two-scale equations Eqs. (14.2) and (14.4).

Application of the two-scale equations to elements of the sets $\Gamma_{[1],0}$ and $\Gamma_{[1],1}$ produces mutually orthogonal pairs of orthonormal sets $\Gamma_{[2],0} \bigoplus \Gamma_{[2],1}$ and $\Gamma_{[2],2} \bigoplus \Gamma_{[2],3}$, respectively, where $\Gamma_{[2],l} \stackrel{\text{def}}{=} \left\{\gamma^{2r}_{[2],l}(\nu)\right\}$, $l = 0, 1, 2, 3$. The linear spans of the orthonormal systems $\Gamma_{[2],l}$ form a split of $\Pi[N]$ into four mutually orthogonal subspaces: $\Pi[N] = \bigoplus_{l=0}^{3} {}^{2r}\mathscr{W}_{[2],l}$. In turn, the system $\Gamma_{[2]} \stackrel{\text{def}}{=} \bigoplus_{l=0}^{3} \Gamma_{[2],l}$ produces the system $\Gamma_{[3]} \stackrel{\text{def}}{=} \bigoplus_{l=0}^{7} \Gamma_{[3],l}$, which generates the split $\Pi[N] = \bigoplus_{l=0}^{7} {}^{2r}\mathscr{W}_{[3],l}$ and so on. Each of the subsequent systems $\Gamma_{[m]}$ forms an orthonormal basis of $\Pi[N]$.

In general, derivation of the "offspring" pair $\Gamma_{[m],2l} \bigoplus \Gamma_{[m],21+1}$ from the "parent" system $\Gamma_{[m-1],l}$ is implemented by the following rules:

If l is even then

$$\begin{aligned}
\gamma^{2r}_{[m],2l}(\nu) &= s_{m-1}(\nu)\, \gamma^{2r}_{[m-1],l}(\nu) + s_{m-1}(\nu + N_m)\, \gamma^{2r}_{[m-1],l}(\nu + N_m), \\
\gamma^{2r}_{[m],2l+1}(\nu) &= d_{m-1}(\nu)\, \gamma^{2r}_{[m-1],l}(\nu) + d_{m-1}(\nu + N_m)\, \gamma^{2r}_{[m-1],l}(\nu + N_m).
\end{aligned} \tag{14.8}$$

If l is odd then

$$\begin{aligned}
\gamma^{2r}_{[m],2l}(\nu) &= d_{m-1}(\nu)\, \gamma^{2r}_{[m-1],l}(\nu) + d_{m-1}(\nu + N_m)\, \gamma^{2r}_{[m-1],l}(\nu + N_m), \\
\gamma^{2r}_{[m],2l+1}(\nu) &= s_{m-1}(\nu)\, \gamma^{2r}_{[m-1],l}(\nu) + s_{m-1}(\nu + N_m)\, \gamma^{2r}_{[m-1],l}(\nu + N_m),
\end{aligned} \tag{14.9}$$

where

$$\begin{aligned}
s_{[m-1]}(\nu) &\stackrel{\text{def}}{=} \sqrt{\frac{U^{4r}_{[m-1]}[\nu]}{2\, U^{4r}_{[m]}[\nu]}}\, \cos^{2r} \frac{\pi \nu}{N_{m-1}}, \\
d_{m-1}(\nu) &\stackrel{\text{def}}{=} \omega^{2^{m-1}\nu}\, s_{m-1}(\nu + N_m).
\end{aligned} \tag{14.10}$$

Similarly to Proposition 14.2, the following shift property can be proved.

Proposition 14.3 *The splines $\gamma^{2r}_{[m],l}(\nu)$ are the eigenvectors of the shift operator*

$$\gamma^{2r}_{[m],l}(\nu)[k + 2^m l] = \omega^{2^m \nu l}\, \gamma^{2r}_{[m],l}(\nu)[k], \quad l = 0, \ldots, N_m - 1. \tag{14.11}$$

The diagram in Fig 14.1 illustrates 3-level orthogonal decomposition of the signal space $\Pi[N]$, which is similar to decomposition of the spline space ${}^{2r}\mathscr{S}$ illustrated in Fig. 8.4. A circle with **s** inside means application of the coefficients $s_{m-1}(\nu)$ to the basis splines, while a circle with **d** inside means application of the coefficients $s_{m-1}(\nu)$.

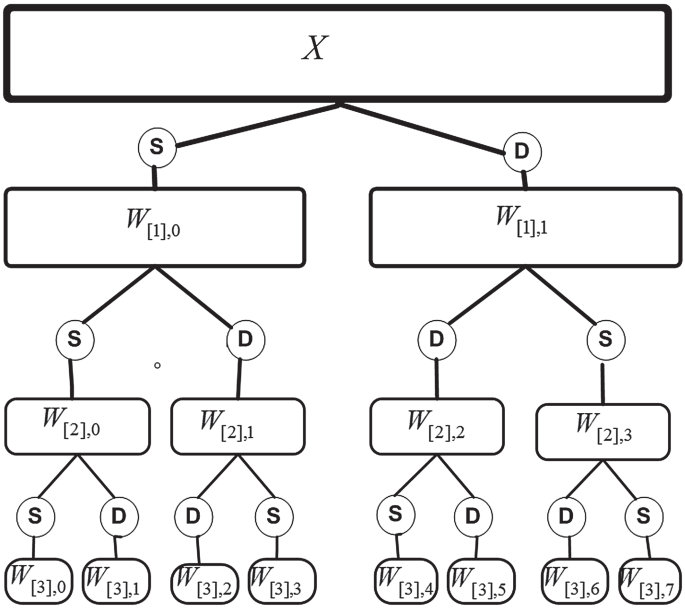


Fig. 14.1 Diagram of 3-level orthogonal decomposition of the signal space

Remark 14.1.1 As in the polynomial splines case, different configurations of the split are possible. An optimal split is implemented via the Best Basis algorithm [1–4], which is discussed in Sect. 8.4.4.

14.2 Discrete Spline Wavelet Packet Transforms

14.2.1 Transform Matrices

As the polynomial spline wavelet packet transforms, the discrete spline wavelet packet transforms can be expressed via the transform matrices. Denote

$$\mathbf{A}_{[m-1]}[\nu] \stackrel{\text{def}}{=} \begin{pmatrix} s_{m-1}(\nu) & s_{m-1}(\nu+N_m) \\ d_{m-1}(\nu) & d_{m-1}(\nu+N_m) \end{pmatrix}, \tag{14.12}$$

where $s_{[m-1]}(\nu)$ and $d_{m-1}(\nu)$ are defined in Eq. (14.10).

Denote, by

$$\mathbf{A}^{\dagger}_{[m-1]}[\nu] \stackrel{\text{def}}{=} \begin{pmatrix} s_{m-1}(\nu) & d^{*}_{m-1}(\nu) \\ s_{m-1}(\nu+N_m) & d^{*}_{m-1}[\nu+N_m], \end{pmatrix} \tag{14.13}$$

the conjugate transpose of the matrix $\mathbf{A}_{[m-1]}[\nu]$, and, by $\mathbf{I}$ the 2×2 identity matrix.

Proposition 14.4 *With any natural m, and* $\nu = -N_m, ..., N_m - 1$, *the matrices* $\mathbf{A}_m[\nu]$ *are unitary, that is*

$$\mathbf{A}_{[m-1]}[\nu] \cdot \mathbf{A}^{\dagger}_{[m-1]}[\nu] = \mathbf{A}^{\dagger}_{m-1}[\nu] \cdot \mathbf{A}_{[m-1]}[\nu] = \mathbf{I}. \tag{14.14}$$

Proof is identical to the proof of Proposition 8.6. Equation (14.14) implies that

Corollary 14.1 *The inverse matrix*

$$\mathbf{A}^{-1}_{[m-1]}[\nu] = \mathbf{A}^{\dagger}_{[m-1]}[\nu] = \begin{pmatrix} s_{m-1}(\nu) & \omega^{-2^{m-1}\nu}\, s_{m-1}(\nu + N_m) \\ s_{m-1}(\nu + N_m) & -\omega^{-2^{m-1}\nu}\, s_{m-1}(\nu) \end{pmatrix}. \tag{14.15}$$

14.2.2 One-Level Transforms

Let a signal $\mathbf{x} = \{x[k]\} \in \Pi[N] = \mathscr{S}_{[0]} = \mathscr{W}_{[0]}$. Denote $\mathbf{s}_{[0]} \stackrel{\text{def}}{=} \mathbf{x}$. It can be represented by the orthonormal basis

$$\mathbf{s}_{[0]} = \sqrt{\frac{1}{N}} \sum_{\nu=-N/2}^{N/2-1} \sigma_{[0]}[\nu] \gamma_{[0]}(\nu)[k], \quad \gamma_{[0]}(\nu)[k] = \frac{\omega^{\nu k}}{\sqrt{N}}, \tag{14.16}$$

$$\sigma_{[0]}[\nu] \stackrel{\text{def}}{=} \hat{x}[\nu] = \sum_{k=0}^{N-1} \omega^{-\nu k}\, x[k] = \sqrt{N}\, \left\langle \mathbf{s}_{[0]}, \gamma_{[0]}(\nu) \right\rangle.$$

The space $\mathscr{W}_{[0]}$ is the orthogonal sum of the subspaces ${}^{2r}\mathscr{W}_{[1],0}$ and ${}^{2r}\mathscr{W}_{[1],1}$ whose orthonormal bases are $\left\{\gamma^{2r}_{[1],0}(\nu)\right\}$ and $\left\{\gamma^{2r}_{[1],1}(\nu)\right\}$, respectively, where $\nu = -N_1/2, ..., N_1/2 - 1$. Thus, $\mathbf{s}_0$ can be represented as the sum of its orthogonal projections onto the subspaces ${}^{2r}\mathscr{W}_{[1],0}$ and ${}^{2r}\mathscr{W}_{[1],1}$: $\mathbf{s}_{[0]} = \mathbf{s}_{[1],0} + \mathbf{s}_{[1],1}$, where

$$\mathbf{s}_{[1],0} \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \sigma_{[1],0}[\nu] \gamma^{2r}_{[1],0}(\nu), \tag{14.17}$$

$$\mathbf{s}_{[1],1} \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{n=-N/4}^{N/4-1} \sigma_{[1],1}[\nu] \gamma^{2r}_{[1],1}(\nu).$$

The orthonormality of the basis splines implies that the coordinates

$$\sigma_{[1],0}[\nu] = \sqrt{N/2} \left\langle \mathbf{s}_{[0]}, \gamma^{2r}_{[1],0}(\nu) \right\rangle, \quad \sigma_{1,1}[\nu] = \sqrt{N/2} \left\langle \mathbf{s}_{[0]}, \gamma^{2r}_{[1],1}(\nu) \right\rangle. \tag{14.18}$$

Using the two-scale relations (14.2) and (14.4), we derive for $\nu = -N/4, ..., N/4-1$

$$\begin{aligned}
\sigma_{[1],0}[\nu] &= \sqrt{\frac{1}{2}}\left(s_0[\nu]\,\sigma_{[0]}[\nu] + s_0[\nu+N/2]\,\sigma_{[0]}[\nu+N/2]\right), \\
\sigma_{[1],1}[\nu] &= \sqrt{\frac{1}{2}}\left(d_0[\nu]\,\sigma[\nu] + d_0[\nu+N/2]\,\sigma_{[0]}[\nu+N/2]\right) \\
\Longleftrightarrow \begin{pmatrix}\sigma_{[1],0}[\nu]\\ \sigma_{[1],1}[\nu]\end{pmatrix} &= \sqrt{\frac{1}{2}}\mathbf{A}_{[0]}[\nu]\cdot\begin{pmatrix}\sigma_{[0]}[\nu]\\ \sigma_{[0]}[\nu+N/2]\end{pmatrix}.
\end{aligned}$$

The inverse relation is

$$\begin{aligned}
\begin{pmatrix}\sigma_{[0]}[\nu]\\ \sigma_{[0]}[\nu+N/2]\end{pmatrix} &= \sqrt{2}\mathbf{A}_0^{\dagger}[\nu]\cdot\begin{pmatrix}\sigma_{[1],0}[\nu]\\ \sigma_{[1],1}[\nu]\end{pmatrix} \\
&= \sqrt{2}\begin{pmatrix}s_0[\nu] & d_0^*[\nu]\\ s_0[\nu+N/2] & d_0^*[\nu+N/2]\end{pmatrix}\cdot\begin{pmatrix}\sigma_{[1],0}[\nu]\\ \sigma_{[1],1}[\nu]\end{pmatrix}, \\
\Longrightarrow \hat{x}[\nu] &= \sigma_{[0]}[\nu] = \sqrt{2}\left(s_0[\nu]\,\sigma_{[1],0}[\nu] + d_0^*[\nu]\,\sigma_{[1],1}[\nu]\right) \qquad (14.19)\\
&= \frac{1}{\sqrt{U_{[1]}^{4r}[\nu]}}\left(\cos^{2r}\frac{\pi\nu}{N}\,\sigma_{[1],0}[\nu] + \omega^{-\nu}\,\sin^{2r}\frac{\pi\nu}{N}\,\sigma_{[1],1}[\nu]\right), \\
\nu &= -N/2, ..., N/2-1.
\end{aligned}$$

Remark 14.2.1 In Eq. (14.19), we used the fact that $U_{[0]}[\nu] = \zeta_{[0]}(\nu)[0] = e^0 = 1$.

In particular, it follows from Eq. (14.19) that the DFT of the splines $\mathbf{s}_{[1],0} \in {}^{2r}\mathscr{W}_{[1],0}$ and $\mathbf{s}_{[1],1} \in {}^{2r}\mathscr{W}_{[1],1}$ presented in Eq. (14.17) are

$$\begin{aligned}
\varsigma_{[1],0}[\nu] &\stackrel{\text{def}}{=} \hat{s}_{[1],0}[\nu] = \sqrt{2}\,s_0[\nu]\,\sigma_{[1],0}[\nu] = \frac{1}{\sqrt{U_{[1]}^{4r}[\nu]}}\cos^{2r}\frac{\pi\nu}{N}\,\sigma_{[1],0}[\nu], \qquad (14.20)\\
\varsigma_{[1],1}[\nu] &\stackrel{\text{def}}{=} \hat{s}_{[1],1}[\nu] = \sqrt{2}\,d_0^*[\nu]\,\sigma_{[1],1}[\nu] = \frac{\omega^{-\nu}}{\sqrt{U_{[1]}^{4r}[\nu]}}\sin^{2r}\frac{\pi\nu}{N}\,\sigma_{[1],1}[\nu].
\end{aligned}$$

14.2.2.1 Remarks on the DFT Spectra

One should keep in mind that the sequences $\left\{U_{[1]}^{4r}[\nu]\right\}$, $\left\{\sigma_{[1],0}[\nu]\right\}$ and $\left\{\sigma_{[1],1}[\nu]\right\}$ are $N/2$-periodic, while the sequences

$$s_0[\nu] = \frac{1}{\sqrt{2\,U_{[1]}^{4r}[\nu]}}\cos^{2r}\frac{\pi\nu}{N}, \quad d_0[\nu] = \frac{\omega^{\nu}}{\sqrt{2\,U_{[1]}^{4r}[\nu]}}\sin^{2r}\frac{\pi\nu}{N}$$

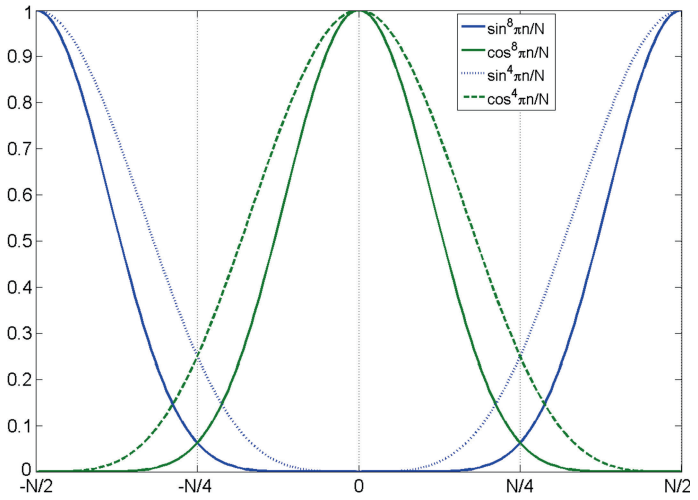


Fig. 14.2 The sequences $\cos^{2r} \pi\nu/N$ and $\sin^{2r} \pi\nu/N$ for $r = 2$ (*dashed curves*) and $r = 4$ (*solid curves*)

are N-periodic. Obviously, the sequence $\cos^{2r} \pi\nu/N$ is essentially supported on the central part $(-N/4, N/4)$ of the interval $(-N/2, N/2)$, which correspond to lower frequencies, while $\sin^{2r} \pi\nu/N$, on the contrary, is supported on the higher frequency region $(-N/2, -N/4) \bigcup (N/4, N/2)$. Thus, multiplication of the $N/2$-periodic sequences $\{\sigma_{[1],i}[\nu]\}$, $i = 0, 1$ with these cosine and sine sequences acts as application of low-pass and high-pass anti-aliasing filters, respectively (Fig. 14.2).

It follows from Eq. (13.13) that

$$2U^{4r}_{[1]}[\nu] = \cos^{4r} \frac{\pi\nu}{N} + \sin^{4r} \frac{\pi\nu}{N}. \tag{14.21}$$

Thus, factor $1/\sqrt{2\,U^{4r}_{[1]}[\nu]}$ makes the shapes of the magnitude responses of the p-filters close to rectangular (Fig. 14.3). The figure was produced by the MATLAB code `plot_s0_d0P`.

Consequently, the DFT spectra of the splines $\mathbf{s}_{[1],0} \in {}^{2r}\mathcal{W}_{[1],0}$ and $\mathbf{s}_{[1],1} \in {}^{2r}\mathcal{W}_{[1],1}$ are concentrated in the low-frequency band $(-N/4, N/4)$ and the high-frequency band $(-N/2, -N/4) \bigcup (N/4, N/2)$, respectively.

14.2.2.2 Periodic Discrete Battle–Lemarié Wavelets of the First Level

Define the signals

$$\psi^{2r}_{[1],0}[k] \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{\nu=-N/4}^{N/4-1} \gamma^{2r}_{[1],0}(\nu)[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \tau_{[1],0}[\nu]\, \omega^{\nu k},$$

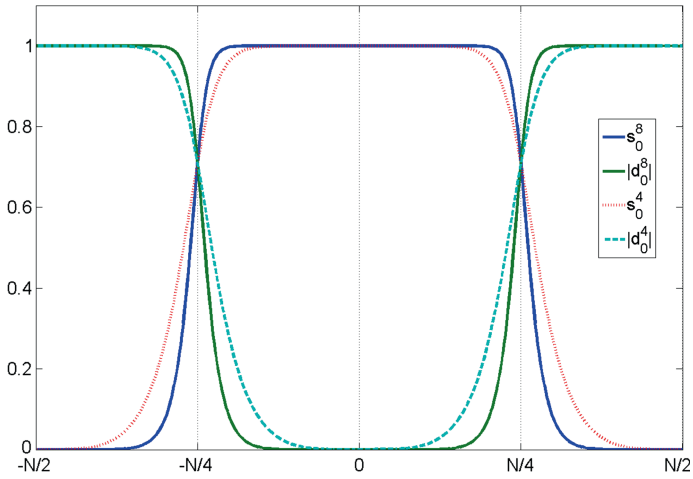


Fig. 14.3 The sequences $s_0^{2r}[\nu]$ and $|d_0^{2r}[\nu]|$ for $r = 2$ (*dashed curves*) and $r = 4$ (*solid curves*)

$$\psi_{[1],1}^{2r}[k] \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \sum_{\nu=-N/4}^{N/4-1} \gamma_{[1],1}^{2r}(\nu)[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \tau_{[1],1}[\nu]\, \omega^{\nu k}.$$

Due to Eq. (14.20), their DFT spectra are

$$\tau_{[1],0}[\nu] = \frac{1}{\sqrt{U_{[1]}^{4r}[\nu]}} \cos^{2r} \frac{\pi \nu}{N} = \sqrt{2}\, s_0[\nu], \tag{14.22}$$

$$\tau_{1,1}[\nu] = \frac{\omega^{-\nu}}{\sqrt{U_{[1]}^{4r}[\nu]}} \sin^{2r} \frac{\pi \nu}{N} = \sqrt{2} d_0^*[\nu].$$

The signal $\psi_{[1],1}^{2r}[k+1]$ has a real-valued DFT spectrum:

$$\psi_{[1],1}^{2r}[k+1] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \bar{\tau}_{[1],1}[\nu]\, \omega^{\nu k},$$

$$\bar{\tau}_{[1],1}[\nu] = \frac{1}{\sqrt{U_{[1]}^{4r}[\nu]}} \sin^{2r} \frac{\pi \nu}{N} = \tau_{[1],0}[\nu + N/2].$$

The discrete splines $\psi_{[1],0}^{2r}$ are called periodic discrete Battle–Lemarié father wavelets of the first level. They coincide with the self-dual generators of the discrete splines spaces defined in Sect. 13.1.5. The discrete splines $\psi_{[1],1}^{2r}$ are called periodic discrete Battle–Lemarié wavelets of the first level. Like the continuous Battle–Lemarié

wavelets defined in Sect. 8.3.2.3, the discrete Battle–Lemarié wavelets form orthonormal bases of the corresponding subspaces.

Proposition 14.5 *The translations* $\left\{\psi^{2r}_{[1],0}[k-2l]\right\}$ *and* $\left\{\psi^{2r}_{[1],1}[k-2l]\right\}$, *where* $l = 0, ..., N/2-1$, *of the first level wavelets form orthonormal bases of the subspaces* ${}^{2r}\mathcal{W}_{[1],0}$ *and* ${}^{2r}\mathcal{W}_{[1],1}$, *respectively.*

Proof The orthonormality and the shift property (14.11) of the discrete exponential splines $\gamma^{2r}_{[1],0}(\nu)$ imply that the inner product

$$\begin{aligned}\sum_{k=0}^{N} \psi^{2r}_{[1],1}[k-2l]\,\psi^{2r}_{[1],1}[k-2\lambda]^* &= \frac{2}{N}\sum_{k=0}^{N}\sum_{\nu,\mu=-N/4}^{N/4-1} \gamma^{2r}_{[1],1}(\nu)[k-2l]\,\gamma^{2r}_{[1],1}(\mu)[k-2\lambda]^* \\ &= \frac{2}{N}\sum_{\nu,\mu=-N/4}^{N/4-1} \omega^{2(l\nu-\lambda\mu)}\sum_{k=0}^{N}\gamma^{2r}_{[1],1}(\nu)[k]\,\gamma^{2r}_{[1],1}(\mu)[k]^* \\ &= \frac{2}{N}\sum_{n=-N/4}^{N/4-1}\omega^{2\nu(l-\lambda)} = \delta[l-\lambda]\,(\mathrm{mod}\,N/2).\end{aligned}$$

Thus, $N/2$ splines $\left\{\psi^{2r}_{[1],1}[k-2l]\right\}$, which are orthogonal to each other and whose norms are equal to one, form an orthonormal basis of the $N/2$-dimensional space ${}^{2r}\mathcal{W}_{[1],1}$. Proof for the splines $\left\{\psi^{2r}_{[1],0}[k-2l]\right\}$ is similar. ■

Figure 14.4, which was prepared by the MATLAB code `D_gen_wp_cons_L1P`, displays the wavelets $\psi^{2r}_{[1],0}$ and $\psi^{2r}_{[1],1}$ for $r = 1, 2, 5$ and their DFT spectra $\tau_{[1],0}[\nu]$ and $|\tau_{[1],1}[\nu]|$, respectively. The spectra $\tau_{[1],0}[\nu]$ occupy the central parts of the plots, which correspond to lower frequencies, while $|\tau_{[1],1}[\nu]|$ are located near the edges, which correspond to higher frequencies. The higher the spline's order is, the closer are the shapes of the spectra to rectangles and the thinner is their overlap. Respectively, the effective support of the wavelets widens when the spline's order increases. Comparing Fig. 14.4 with Figs. 8.3 and 9.2, we observe that the shapes of the discrete wavelets and of their spectra are very similar to the shapes of their continuous counterparts and of their SHA spectra and also to the shapes of corresponding discrete–time wavelets described in Sect. 9.1.2.

14.2.3 Multilevel Wavelet Packet Transforms

14.2.3.1 Transforms of the Spline Coordinates

Assume that a discrete spline $\mathbf{s}_{[m-1],l} \in {}^{2r}\mathcal{W}_{[m-1],l}$ is decomposed into its orthogonal projections $\mathbf{s}_{[m],2l} \in {}^{2r}\mathcal{W}_{[m],2l}$ and $\mathbf{s}_{[m],2l+1} \in {}^{2r}\mathcal{W}_{[m],2l+1}$

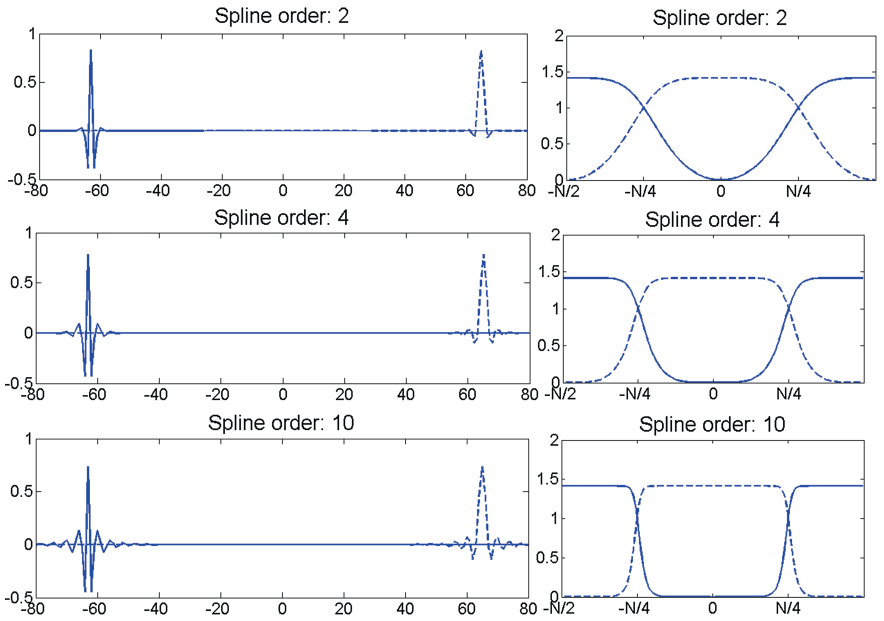


Fig. 14.4 *Left* Periodic discrete Battle–Lemarié wavelets $\psi^{2r}_{[1],0}$ (*dashed curves*) and $\psi^{2r}_{[1],1}$ (*solid curves*), $r = 1, 2, 5$, of the first decomposition level. *Right* Their DFT spectra $\tau_{[1],0}[\nu]$ (*dashed curves*), and $|\tau_{[1],1}[\nu]|$ (*solid curves*), $N = 256$

$$\mathbf{s}_{[m-1],l} = \sqrt{\frac{1}{N_{m-1}}} \sum_{\nu=-N_{m-1}/2}^{N_{m-1}/2-1} \sigma_{[m-1],l}[\nu]\, \gamma^{2r}_{[m-1],l}[\nu], \tag{14.23}$$

$$\mathbf{s}_{[m],2l+\lambda} = \sqrt{\frac{1}{N_m}} \sum_{\nu=-N_m/2}^{N_{m-1}/2-1} \sigma_{[m],2l+\lambda}[\nu]\, \gamma^{2r}_{[m],2l+\lambda}[\nu], \quad \lambda = 0, 1.$$

Assume that the index l is even. Similarly to the first decomposition level, we derive from Eq. (14.8) the following relation between the coordinates when $\nu = -N_m/2, \ldots, N_m/2 - 1$

$$\begin{pmatrix} \sigma_{[m],2l}[\nu] \\ \sigma_{[m],2l+1}[\nu] \end{pmatrix} = \sqrt{\frac{1}{2}} \mathbf{A}_{[m-1]}[\nu] \cdot \begin{pmatrix} \sigma_{[m-1],l}[\nu] \\ \sigma_{[m-1],l}[\nu + N_m] \end{pmatrix}. \tag{14.24}$$

The inverse relation is

$$\begin{pmatrix} \sigma_{[m-1],l}[\nu] \\ \sigma_{[m-1],l}[\nu + N_m] \end{pmatrix} = \sqrt{2} \mathbf{A}^{\dagger}_{[m-1]}[\nu] \cdot \begin{pmatrix} \sigma_{[m],2l}[\nu] \\ \sigma_{[m],2l+1}[\nu] \end{pmatrix}, \tag{14.25}$$

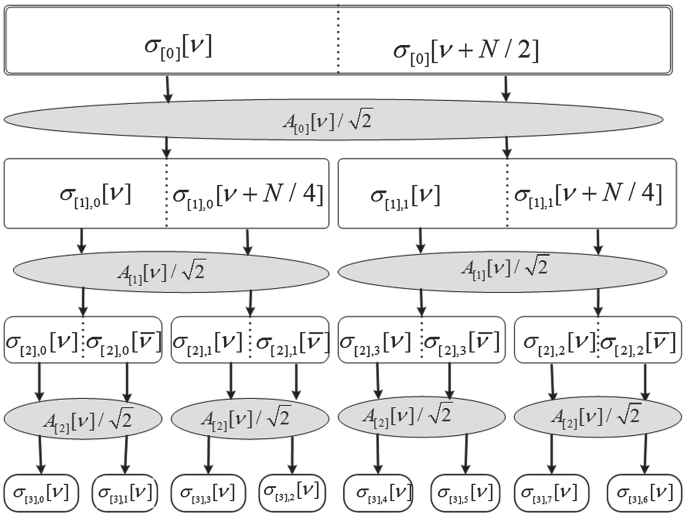


Fig. 14.5 Diagram of 3-level discrete wavelet packet transform of a signal $\mathbf{x} \in \Pi[N]$, whose DFT spectrum is $\{\hat{x}[\nu] = \sigma_{[0]}[\nu]\}$. Notation $\bar{\nu} \stackrel{\text{def}}{=} \nu + N/8$

where the matrices $\mathbf{A}_{[m-1]}[\nu]$ and $\mathbf{A}^{\dagger}_{[m-1]}[\nu]$ are defined in Eqs. (14.12) and (14.13), respectively.

If the index l is odd then for $\nu = -N_m/2, ..., N_m/2 - 1$,

$$\begin{pmatrix} \sigma_{[m],2l+1}[\nu] \\ \sigma_{[m],2l}[\nu] \end{pmatrix} = \sqrt{\frac{1}{2}}\mathbf{A}_{[m-1]}[\nu] \cdot \begin{pmatrix} \sigma_{[m-1],l}[\nu] \\ \sigma_{[m-1],l}[\nu + N_m] \end{pmatrix}, \tag{14.26}$$

$$\begin{pmatrix} \sigma_{[m-1],l}[\nu] \\ \sigma_{[m-1],l}[\nu + N_m] \end{pmatrix} = \sqrt{2}\mathbf{A}^{\dagger}_{[m-1]}[\nu] \cdot \begin{pmatrix} \sigma_{[m],2l+1}[\nu] \\ \sigma_{[m],2l}[\nu] \end{pmatrix}, \tag{14.27}$$

The direct wavelet packet transform of a signal or a set of signals is implemented by the MATLAB function `D_WP_analP.m`.

The diagram of the 3-level discrete wavelet packet transform of a signal $\mathbf{x} \in \Pi[N]$, whose DFT spectrum is $\{\hat{x}[\nu] = \sigma_{[0]}[\nu]\}$ is displayed in Fig. 14.5. For compactness, we denote $\bar{\nu} \stackrel{\text{def}}{=} \nu + N/8$.

The reconstruction is implemented in an inverse order.

14.2.3.2 Computation of a Discrete Spline Belonging to ${}^{2r}\mathscr{W}_{[m],l}$

Assume that a discrete spline $\mathbf{s}_{[m],l}$ belongs to the subspace ${}^{2r}\mathscr{W}_{[m],l} \subset \pi[N]$. Then, it is represented as

$$s_{[m],l}[k] = \frac{1}{\sqrt{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \sigma_{[m],l}[\nu]\, \gamma^{2r}_{[m],l}(\nu)[k]. \tag{14.28}$$

One way to calculate its DFT spectrum is explained in the following remark.

Remark 14.2.2 In order to derive the DFT spectrum $\left\{\varsigma_{[m],l}[\nu] = \hat{s}_{[m],l}[\nu]\right\}$, $\nu = -N/2, ..., N/2-1$, of a discrete spline $\mathbf{s}_{[m],l}$, one should put $\sigma_{[m],\tilde{l}}[\nu] = 0$ for all $\nu = -N_m/2, ..., N_m/2-1$ and $\tilde{l} \neq l$. Then, the reconstruction operations (14.25) or (14.27) should be iteratively implemented starting from the level m till the initial level.

As a result, we get

$$\varsigma_{[m],l}[\nu] = \hat{s}_{[m],l}[\nu] = F_{[m],l}[\nu]\, \sigma_{[m],l}[\nu], \;\; \nu = -N/2, ..., N/2-1,$$

where multiplication by the sequence $F_{[m],l}[\nu]$ acts as antialiasing p-filtering of the N_m-periodic sequence $\sigma_{[m],l}[\nu]$.

Example

The splines of the second decomposition level are

$$\mathbf{s}_{[2],l}[k] = \frac{1}{\sqrt{N/4}} \sum_{\nu=-N/8}^{N/8-1} \sigma_{[2],l}[\nu]\, \gamma^{2r}_{[2],l}(\nu)[k], \quad l = 0, 1, 2, 3.$$

$$\varsigma_{[2],l}[\nu] = F_{[2],l}[\nu]\, \sigma_{[2],l}[\nu], \;\; \nu = -N/2, ..., N/2-1, \tag{14.29}$$

$$F_{[2],0}[\nu] = 2\, s_0[\nu]\, s_1[\nu] = \sqrt{\frac{1}{U^{4r}_{[2]}[\nu]}}\, \cos^{2r}\frac{\pi\,\nu}{N}\, \cos^{2r}\frac{2\pi\,\nu}{N},$$

$$F_{[2],1}[\nu] = 2\, s_0[\nu]\, d^*_1[\nu] = \omega^{-2\nu} \sqrt{\frac{U^{4r}_{[1]}[\nu + N/4]}{U^{4r}_{[1]}[\nu]\, U^{4r}_{[2]}[\nu]}}\, \cos^{2r}\frac{\pi\,\nu}{N}\, \sin^{2r}\frac{2\pi\,\nu}{N}$$

$$F_{[2],3}[\nu] = 2\, d^*_0[\nu]\, s_1[\nu] = \omega^{-\nu} \sqrt{\frac{1}{U^{4r}_{[2]}[\nu]}}\, \sin^{2r}\frac{\pi\,\nu}{N}\, \cos^{2r}\frac{2\pi\,\nu}{N},$$

$$F_{[2],2}[\nu] = 2\, d^*_0[\nu]\, d^*_1[\nu] = \omega^{-3\nu} \sqrt{\frac{U^{4r}_{[1]}[\nu + N/4]}{U^{4r}_{[1]}[\nu]\, U^{4r}_{[2]}[\nu]}}\, \sin^{2r}\frac{\pi\,\nu}{N}\, \sin^{2r}\frac{2\pi\,\nu}{N}.$$

The magnitude responses of the p-filters $F_{[2],l}[\nu]$ are displayed in Fig. 14.6 (MATLAB code `D_pfilt_L2P.m`). Figure 14.7 (MATLAB code `D_pfilt_L3P.m`) displays the magnitude responses of the third level p-filters $F_{[3],l}[\nu]$. For better resolution of the picture, in Fig. 14.7 the magnitude responses are displayed for the half-band $n = 0, ..., N/2-1$. The other half-band $n = -N/2, ..., -1$ mirrors

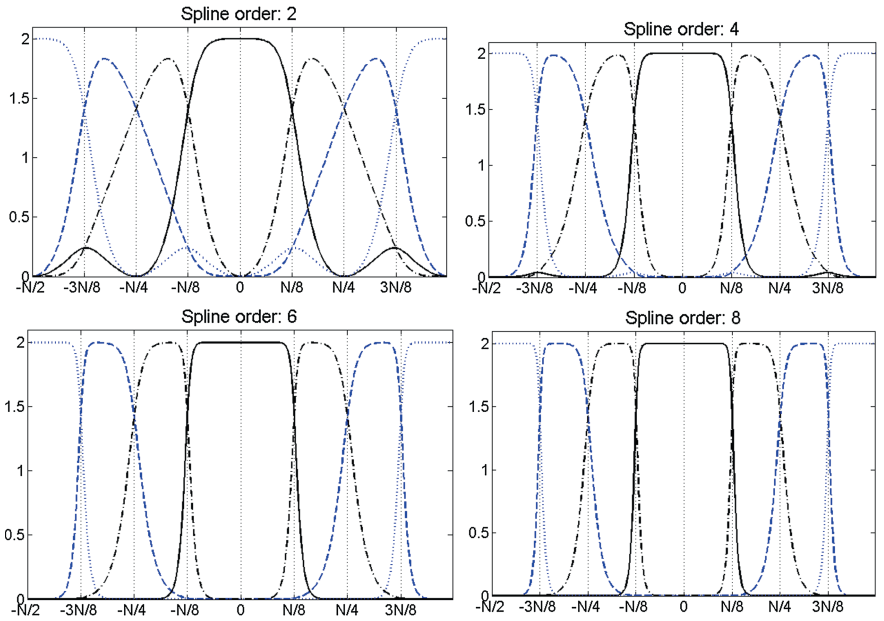


Fig. 14.6 The magnitude responses of the second level p-filters $F_{[2],l}[\nu]$ (*solid curves*), $\left|F_{[2],1}[\nu]\right|$ (*dash-dot curves*), $\left|F_{[2],2}[\nu]\right|$ (*dashed curves*) and $\left|F_{[2],3}[\nu]\right|$ (*dotted curves*) for the splines of orders 2, 4, 6, 8, $N = 256$

the former one. One can observe that shapes of these magnitude responses is similar to shapes of the respective p-filters arising in the polynomial spline wavelet packet transforms (Figs. 8.5 and 8.6).

14.3 Discrete Spline Wavelet Packets

Once we have the exponential discrete splines $\gamma^{2r}_{[m],l}(\nu)$, which form orthonormal bases of the subspaces ${}^{2r}\mathscr{W}_{[m],l} \subset \Pi[N]$, the discrete spline wavelet packets $\psi^{2r}_{[m],l} \in {}^{2r}\mathscr{W}_{[m],l}$ are defined similarly to the continuous spline wavelet packets:

$$\psi^{2r}_{[m],l}[k] \stackrel{\text{def}}{=} \sqrt{\frac{1}{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \gamma^{2r}_{[m],l}(\nu)[k] \in {}^{2r}\mathscr{W}_{[m],l}. \tag{14.30}$$

As in the continuous case, the following claims are true.

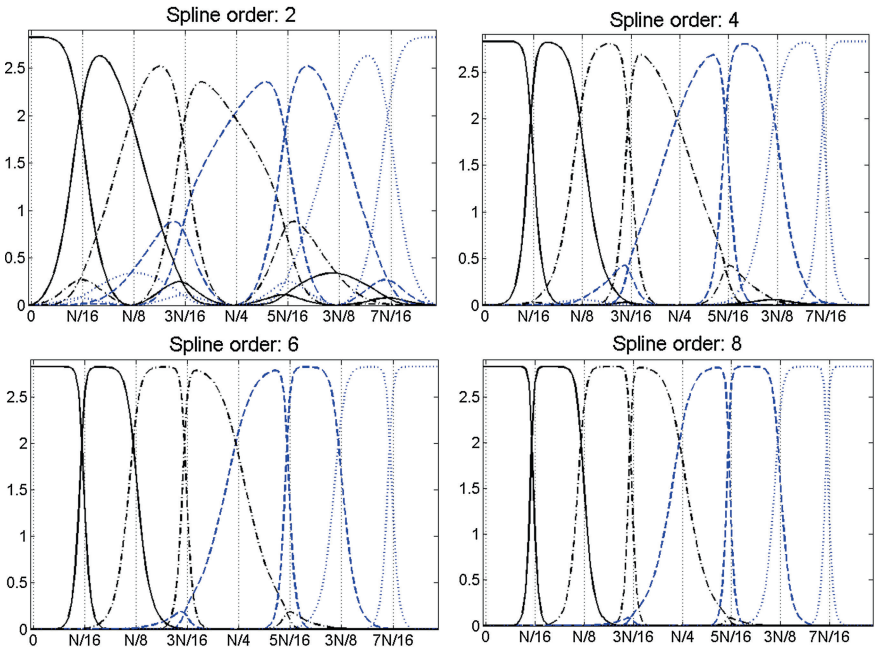


Fig. 14.7 The magnitude responses of the third level p-filters $F_{[3],l}[\nu]$, $l = 0, \ldots, 7$, for the splines of orders 2, 4, 6, 8, $N = 256$

Proposition 14.6 *There holds the relation reciprocal to Eq.* (14.30)*:*

$$\gamma^{2r}_{[m],l}(\nu)[k] = \sqrt{\frac{1}{N_m}} \sum_{\kappa=0}^{N_m-1} \omega^{2^m \kappa \nu}\, \psi^{2r}_{[m],l}[k - 2^m \kappa]. \tag{14.31}$$

Proof Equation (14.30) and the shift property Eq. (14.11) imply that

$$\psi^{2r}_{[m],l}[k - 2^m \kappa] = \sqrt{\frac{1}{N_m}} \sum_{n=-N_m/2}^{N_m/2-1} \omega^{-2^m kn}\, \gamma^{2r}_{[m],l}(\nu)[k].$$

Applying the IDFT, we get Eq. (14.31). ■

Proposition 14.7 *The translations* $\left\{\psi^{2r}_{[m],l}[k - 2^m \lambda]\right\}$, $\lambda = 0, \ldots, N_m - 1$, *where* $N_m = N/2^m$, *form an orthonormal bases for* ${}^{2r}\mathcal{W}_{[m],l}$. *The union*

$$\bigcup_{l=0}^{2^m-1} \left\{\psi^{2r}_{[m],l}[k - 2^m \lambda]\right\}, \quad \lambda = 0, \ldots, N_m - 1,$$

forms an orthonormal basis for the entire space $\Pi[N]$.

Proof The spline $\psi^{2r}_{[m],l}[k-2^m\lambda]$ is orthogonal to any spline $\psi^{2r}_{[m],\tilde{l}}[k-2^m\mu]$, for $\tilde{l}\neq l$ because they belong to mutually orthogonal subspaces. The inner product of two splines from the same subspace is

$$\begin{aligned}
&\sum_{k=0}^{N}\psi^{2r}_{[m],l}[k-2^m\lambda]\,\psi^{2r}_{[m],l}[k-2^m\mu]^*\\
&=\frac{1}{N_m}\sum_{\nu,\tilde{\nu}=-N_m/2}^{N_m/2-1}\omega^{-2^m(\lambda\nu-\tilde{l}a\tilde{\nu})}\sum_{k=0}^{N}\gamma^{2r}_{[m]}(\nu)[k]\gamma^{2r}_{m}[\tilde{\nu}]^*[k]\\
&=\frac{1}{N_m}\sum_{\nu=-N_m/2}^{N_m/2-1}w^{-2^m\nu(\lambda-\tilde{\lambda})}=\delta[\lambda-\tilde{\lambda}],\quad \lambda=0,\ldots,N_m-1.
\end{aligned}$$

Thus, N_m mutually orthogonal discrete splines $\left\{\psi^{2r}_{[m],l}[k-2^m\lambda]\right\}$, $\lambda=0,\ldots,N_m-1$, whose norms are equal to one, form a basis of the N_m-dimensional space ${}^{2r}\mathscr{W}_{[m],l}$. ■

The splines $\psi^{2r}_{[m],0}[k]$ and $\psi^{2r}_{[m],1}[k]$ are periodic discrete Battle–Lemarié father and mother wavelets, respectively. The splines $\psi^{2r}_{[m],l}[k]$ with arbitrary $l=1,\ldots,2^m-1$, are called discrete spline periodic orthonormal wavelet packets. Apparently, the single wavelet packet in the initial space $\mathscr{S}_{[0]}=\Pi[N]$ is the N-periodic delta sequence $\psi_{[0]}[k]=\delta[k]$.

14.3.1 The DFT Spectra of Spline Wavelet Packets

To efficiently operate with the discrete spline wavelet packets $\psi^{2r}_{[m],l}[k]$, we need to know the DFT spectra of these wavelet packets

$$\tau_{[m],l}[\nu]\stackrel{\text{def}}{=}\sum_{k=0}^{N-1}\omega^{-k\nu}\,\psi^{2r}_{[m],l}[k]\Longleftrightarrow\psi^{2r}_{[m],l}[k]=\frac{1}{N}\sum_{\nu=-N/2}^{N/2-1}\omega^{k\nu}\,\tau_{[m],l}[\nu]. \tag{14.32}$$

At the initial scale $\mathscr{S}_{[0]}=\Pi[N]$,

$$\psi_{[0]}[k]=\delta[k]=N^{-1}\sum_{\nu=-N/2}^{N/2-1}\omega^{k\nu}\Longrightarrow\tau_{[0]}[\nu]=1.$$

The spectra of the first level wavelet packets $\psi^{2r}_{[1],l}[k]$, $l=0.1$, were presented in Eq. (14.22)

$$\tau_{[1],0}[\nu]=\frac{1}{\sqrt{U^{4r}_{[1]}[\nu]}}\cos^{2r}\frac{\pi\nu}{N}=\sqrt{2}\,s_0[\nu],\quad \tau_{1,1}[\nu]=\frac{\omega^{-\nu}}{\sqrt{U^{4r}_{[1]}[\nu]}}\sin^{2r}\frac{\pi\nu}{N}=\sqrt{2}d^*_0[\nu].$$

Generally, in line with Remark 14.2.2, to derive the DFT spectrum $\left\{\tau_{[m],l}[\nu]\right\}$, $\nu=-N/2,\ldots,N/2-1$, of a wavelet packet $\psi^{2r}_{[m],l}[k]$, one should put $\sigma_{[m],l}[\nu]=1$, and $\sigma_{[m],\tilde{l}}[\nu]=0$ for all $\nu=-N_m/2,\ldots,N_m/2-1$ and $\tilde{l}\neq l$. Then, the reconstruction operations (14.25) should be iteratively implemented, starting from the level m till the initial level. However, practically, the calculations can be implemented by the following tree-structured scheme.

Proposition 14.8 *The DFT spectra of discrete spline wavelet packets are derived iteratively by the following rule:* $\tau_{[0]}[\nu]=1$,

$$\begin{aligned}\begin{pmatrix}\tau_{[m],2l}[\nu]\\ \tau_{[m],2l+1}[\nu]\end{pmatrix}&=\sqrt{2}\,\tau_{[m-1],l}[\nu]\begin{pmatrix}s_{m-1}(\nu)\\ d^*_{m-1}[\nu]\end{pmatrix}\ \textit{if } l \textit{ is even},\\ \begin{pmatrix}\tau_{[m],2l}[\nu]\\ \tau_{[m],2l+1}[\nu]\end{pmatrix}&=\sqrt{2}\,\tau_{[m-1],l}[\nu]\begin{pmatrix}d^*_{m-1}[\nu]\\ s_{m-1}(\nu)\end{pmatrix}\ \textit{if } l \textit{ is odd},\end{aligned}\tag{14.33}$$

where $\nu=-N/2,\ldots N/2-1$.

Proof Assume that l is even. Then, using Eq. (14.8), we have

$$\begin{aligned}\tau_{[m],2l}[\nu]&=\sum_{k=0}^{N-1}\omega^{-k\nu}\,\psi^{2r}_{[m],2l}[k]=\sqrt{\frac{2^m}{N}}\sum_{k=0}^{N-1}\omega^{-k\nu}\sum_{\mu=-N_m/2}^{N_m/2-1}\gamma^{2r}_{[m],2l}(\mu)[k]\\&=\sqrt{\frac{2^m}{N}}\sum_{k=0}^{N-1}\omega^{-k\nu}\sum_{\mu=0}^{N_m-1}\Big(s_{m-1}[\mu]\,\gamma^{2r}_{[m-1],l}(\mu)[k]\\&\quad+s_{m-1}[\mu+N_m]\,\gamma^{2r}_{[m-1],l}(\mu+N_m)[k]\Big)\\&=\sqrt{2}\sqrt{\frac{2^{m-1}}{N}}\sum_{k=0}^{N-1}\omega^{-k\nu}\sum_{\nu=0}^{N_{m-1}-1}s_{m-1}[\mu]\,\gamma^{2r}_{[m-1],l}(\mu)[k]\end{aligned}$$

Equation (14.31) implies that

$$\gamma^{2r}_{[m-1],l}(\mu)[k]=\sqrt{\frac{1}{N_{m-1}}}\sum_{\kappa=0}^{N_{m-1}-1}\omega^{2^{m-1}\kappa\mu}\,\psi^{2r}_{[m-1],l}[k-2^{m-1}\kappa].$$

Consequently,

$$\tau_{[m],2l}[\nu]=\sqrt{2}\,\frac{1}{N_{m-1}}\sum_{k=0}^{N-1}\omega^{-k\nu}\sum_{\mu=0}^{N_{m-1}-1}s_{m-1}[\mu]$$

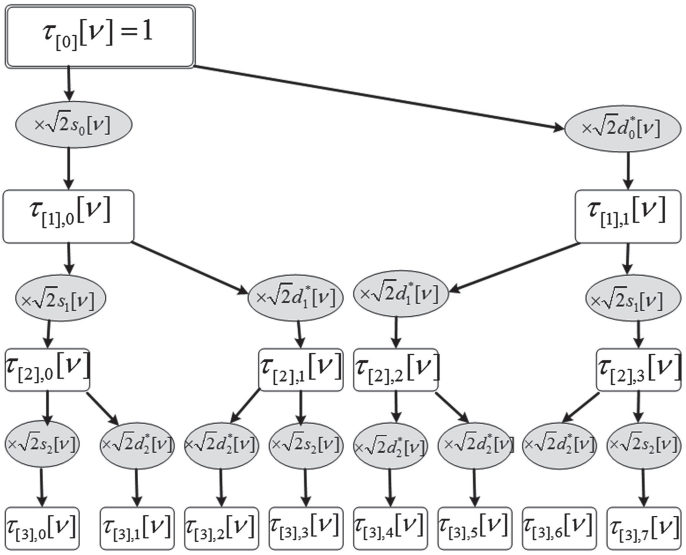


Fig. 14.8 Diagram of calculation of the DFT spectra of 3-level discrete spline wavelet packets

$$
\begin{aligned}
&\times \sum_{\kappa=0}^{N_{m-1}-1} \omega^{2^{m-1}\kappa\mu}\, \psi^{2r}_{[m-1],l}[k-2^{m-1}\kappa] \\
&= \sqrt{2}\,\frac{1}{N_{m-1}} \sum_{\mu=0}^{N_{m-1}-1} s_{m-1}[\mu] \sum_{\kappa=0}^{N_{m-1}-1} \omega^{2^{m-1}\kappa\mu} \sum_{k=0}^{N-1} \omega^{-(k+2^{m-1}\kappa)\nu}\, \psi^{2r}_{[m-1],l}[k] \\
&= \sqrt{2}\,\tau_{[m],l}[\nu]\,\frac{1}{N_{m-1}} \sum_{\mu=0}^{N_{m-1}-1} s_{m-1}[\mu] \sum_{\kappa=0}^{N_{m-1}-1} \omega^{2^{m-1}\kappa(\mu-\nu)} \\
&= \sqrt{2}\,\tau_{[m],l}[\nu]\, s_{m-1}[\nu].
\end{aligned}
$$

The rest of relations in Eq. (14.33) are proved similarly. ■

Remark 14.3.1 Like continuous wavelet packets, the set of the discrete spline wavelet packets has a tree structure. The wavelet packet $\psi^{2r}_{[m-1],l}$ is a "parent" of $\psi^{2r}_{[m],2l}$ and $\psi^{2r}_{[m],2l+1}$ while $\psi^{2r}_{[m],2l}$ and $\psi^{2r}_{[m],2l+1}$ are its "offsprings".

The DFT spectra of the discrete spline wavelet packets are calculated by the MATLAB function `D_wP_coefP.m`. The scheme is illustrated by the diagram in Fig. 14.8, which is similar to the diagram of the scheme for calculation of the SHA spectra of the continuous wavelet packets displayed in Fig. 8.7.

Example:

The DFT spectra of wavelet packets from the second level are $\tau_{[2],l}[\nu] = F_{[2],l}[\nu]$, where $F_{[2],l}[\nu]$ are the frequency responses of p-filters, that are presented in Eqs.

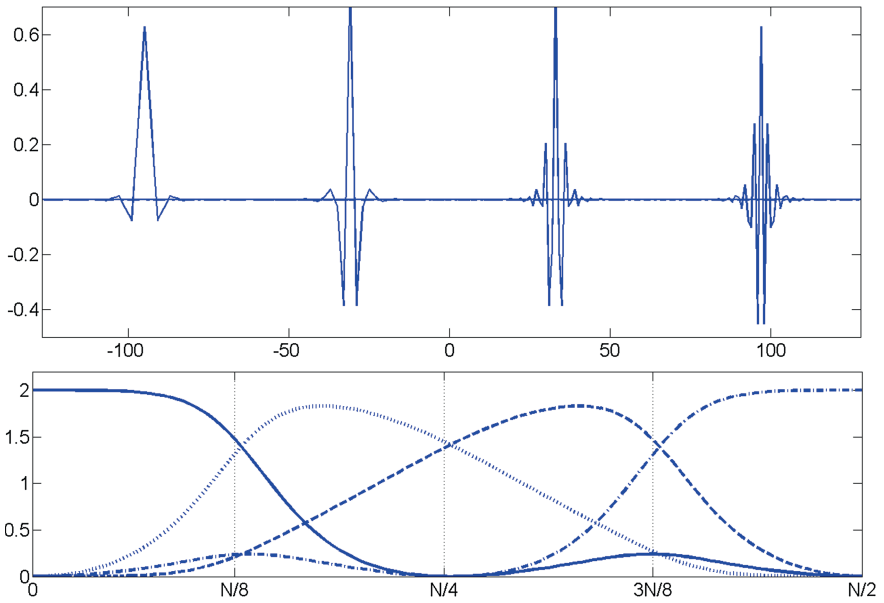


Fig. 14.9 Second level discrete spline wavelet packets and their DFT spectra. *Upper frame* from *left* to *right* $\psi^2_{[2],l}[k]$, $l = 0, 1, 2, 3$. *Bottom frame* $\tau_{[2],0}[\nu]$, (*solid line*) $\tau_{[2],l}[\nu]$ (*dotted*), $\tau_{[2],2}[\nu]$ (*dashed*), $\tau_{[2],3}[\nu]$ (*dash-dotted*)

(14.29). Figures 14.9, 14.10 and 14.11, which are produced by the MATLAB code `D_wp_level2P`, display the second level wavelet packets and magnitudes of their DFT spectra for the splines of orders 2, 4 and 8, respectively. Only the right half band $[0, N/2]$ of the spectra is displayed.

14.3.2 Discrete Splines Wavelet Packets Bases

Assume that a discrete spline $\mathbf{s}$ belongs to ${}^{2r}\mathscr{W}_{[m],l}$. Then, it can be expanded over the orthonormal wavelet packets basis. Equation (14.30) and the shift property (14.3) imply that

$$s[k] = \sum_{\lambda=0}^{N_m-1} q_{[m],l}[\lambda]\, \psi^{2r}_{[m],l}[k - 2^m\lambda] \tag{14.34}$$

$$= \sqrt{\frac{1}{N_m}} \sum_{\lambda=0}^{N_m-1} q_{[m],l}[\lambda] \sum_{\nu=-N_m/2}^{N_m/2-1} \gamma^{2r}_{[m],l}(\nu)[k - 2^m\lambda]$$

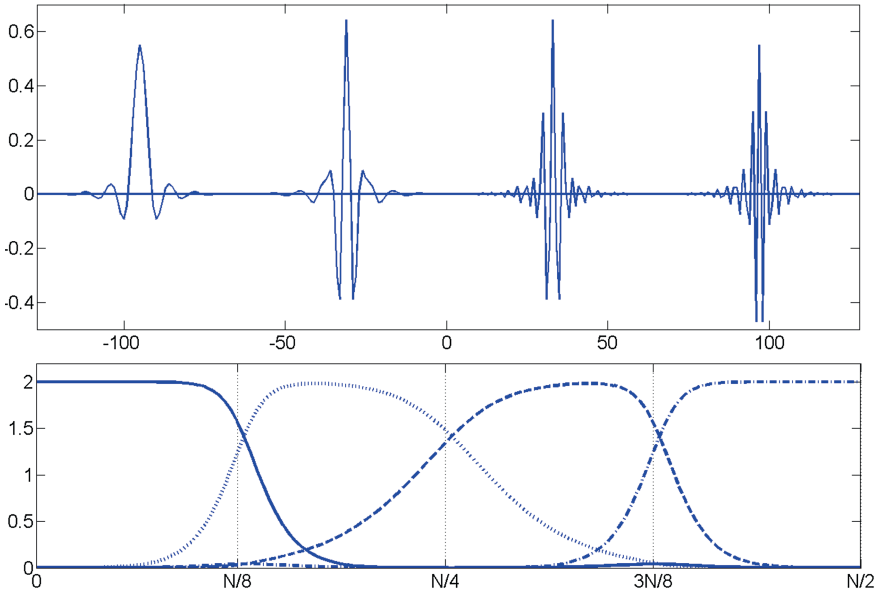


Fig. 14.10 Second level discrete spline wavelet packets and their DFT spectra. *Upper frame* from *left* to *right* $\psi^4_{[2],l}[k]$, $l = 0, 1, 2, 3$. *Bottom frame* $\tau_{[2],0}[\nu]$, (*solid line*) $\tau_{[2],l}[\nu]$ (*dotted*), $\tau_{[2],2}[\nu]$ (*dashed*), $\tau_{[2],3}[\nu]$ (*dash-dotted*)

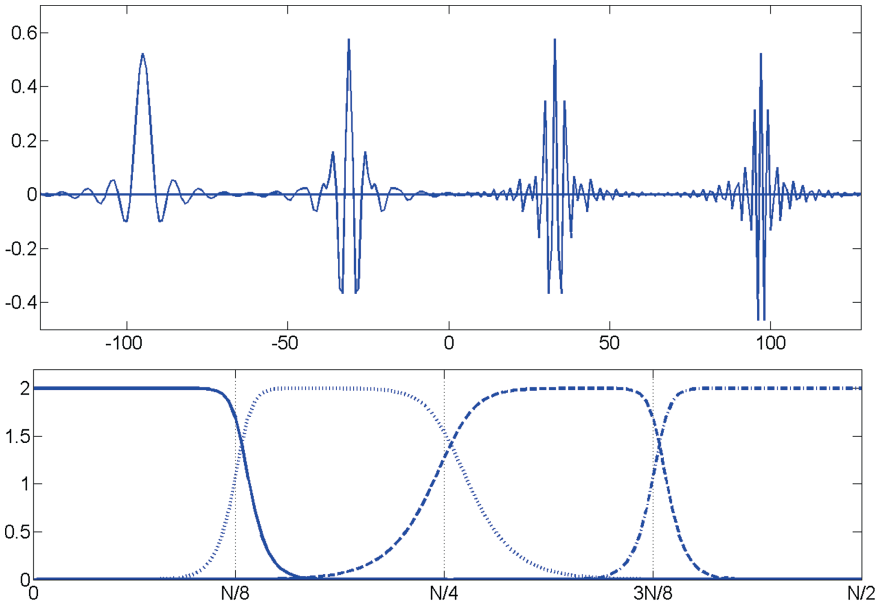


Fig. 14.11 Second level discrete spline wavelet packets and their DFT spectra. *Upper frame* from *left* to *right* $\psi^8_{[2],l}[k]$, $l = 0, 1, 2, 3$. *Bottom frame* $\tau_{[2],0}[\nu]$, (*solid line*) $\tau_{[2],l}[\nu]$ (*dotted*), $\tau_{[2],2}[\nu]$ (*dashed*), $\tau_{[2],3}[\nu]$ (*dash-dotted*)

$$= \sqrt{\frac{1}{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \gamma_{[m],l}^{2r}(\nu)[k] \sum_{\lambda=0}^{N_m-1} \omega^{-2^m \nu\lambda}\, q_{[m],l}[\lambda]$$

$$= \sqrt{\frac{1}{N_m}} \sum_{\nu=-N_m/2}^{N_m/2-1} \sigma_{[m],l}[\nu]\, \gamma_{[m],l}^{2r}(\nu)[k].$$

Thus, coordinates of the spline $\mathbf{s} \in {}^{2r}\mathscr{W}_{[m],l}$ in two orthonormal bases of the subspace ${}^{2r}\mathscr{W}_{[m],l} \subset \Pi[N]$ are linked via the DFT as

$$\sigma_{[m],l}[\nu] = \sum_{\lambda=0}^{N_m-1} \omega^{-2^m \nu\lambda}\, q_{[m],l}[\lambda], \tag{14.35}$$

$$q_{[m],l}[\lambda] = \frac{1}{N_m} \sum_{\nu=-N_m/2}^{N_m/2-1} \omega^{2^m \nu\lambda}\, \sigma_{[m],l}[\nu].$$

The DFT of the discrete spline $\mathbf{s} \in {}^{2r}\mathscr{W}_{[m],l}$, which is represented by Eq. (14.34), is

$$\hat{s}[n] = \sum_{k=0}^{N-1} \omega^{-nk} \sum_{\lambda=0}^{N_m-1} q_{[m],l}[\lambda]\, \psi_{[m],l}^{2r}[k - 2^m\lambda] \tag{14.36}$$

$$= \sum_{\lambda=0}^{N_m-1} q_{[m],l}\, \omega^{-n2^m\lambda} \sum_{k=0}^{N-1} \omega^{-nk}\, \psi_{[m],l}^{2r}[k] = \hat{q}_{[m],l}[n]_m\, \hat{\psi}_{[m],l}^{2r}[n]$$

$$= \hat{q}_{[m],l}[n]_m\, \tau_{[m],l}[n].$$

Assume that the DFT of a signal $\mathbf{x} = \{x[k]\} \in \Pi[N]$ is $\left\{\hat{x}[\nu] = \sigma_{[0]}[\nu]\right\}$. Then, all the coordinates $\sigma_{[m],l}[\nu]$ of the orthogonal projections $\mathbf{s}_{[m],l}$ of the signal $\mathbf{x}$ onto the subspaces ${}^{2r}\mathscr{W}_{[m],l}$, $m = 1, ..., M$, $l = 1, ..., 2^m - 1$ are calculated by simple arithmetic operations (see Eqs. (14.24) and (14.26) and Fig. 14.5). Then, by utilizing Eqs. (14.35), we find the coordinates $q_{[m],l}[\lambda]$ of the splines $\mathbf{s}_{[m],l}$ in the discrete spline wavelet packets bases $\left\{\psi_{[m],l}^{2r}[k - 2^m\lambda]\right\}$. These computations consist of forward and backward application of the fast Fourier transform (FFT). The discrete spline wavelet packet transform of a signal or of an array consisting of row signals is implemented by the MATLAB function `D_WP_analP.m`.

Once these coordinates are computed, the signal $\mathbf{x}$ can be represented via a variety of orthonormal bases, which are constituted by the wavelet packets $\left\{\psi_{[m],l}^{2r}[k - 2^m\lambda]\right\}$ that belong to different combinations of the subspaces ${}^{2r}\mathscr{W}_{[m],l}$. Some of the options are:

"Horizontal" bases

"Horizontal" bases consist of shifts of all the wavelet packets from the same level m, where m can be one of the integers 1, ..., M:

$$\mathbf{B}^{h}_{[m]} \stackrel{\text{def}}{=} \bigcup_{l=0}^{2^m-1} \left\{ \psi^{2r}_{[m],l}[k - 2^m\lambda] \right\}, \tag{14.37}$$

where $\lambda = 0, ..., N_m - 1$, $N_m = N/2^m$. In this case, the DFT spectrum of a signal $\mathbf{x}$ is split into 2^m regions of approximately the same size as it is seen, for example, in Figs. 14.9, 14.10 and 14.11. Restoration of a signal or of an array consisting of row signals from its representation in the "Horizontal" basis is implemented by the MATLAB function `D_hor_WP_synthP.m`.

Wavelet bases

$$\mathbf{B}^{w}_{[m]} \stackrel{\text{def}}{=} \left\{ \psi^{2r}_{[m],0}[k - 2^m\lambda] \right\}_{\lambda=0}^{N_m-1} \bigcup \left\{ \psi^{2r}_{[m],1}[k - 2^m\lambda] \right\}_{\lambda=0}^{N_m-1} \tag{14.38}$$
$$\bigcup \left\{ \psi^{2r}_{[m]-1,1}[k - 2^{m-1}\lambda] \right\}_{\lambda=0}^{N_{m-1}-1} \bigcup ... \bigcup \left\{ \psi^{2r}_{1,1}[k - 2\lambda] \right\}_{\lambda=0}^{N/2-1}.$$

The discrete spline wavelet decomposition of a signal or of an array consisting of row signals is implemented by the MATLAB function `D_wav_analP.m`. Reconstruction is implemented by the MATLAB function `D_wav_syntlP.m`.

The DFT spectrum of a signal $\mathbf{x}$ is split in a logarithmic mode. Figures 14.12, 14.13 and 14.14, which are produced by the MATLAB code `D_wW_level3P.m`, display wavelets $\psi^{2r}_{[3],0}[k]$, $\psi^{2r}_{[3],1}[k]$, $\psi^{2r}_{[2],1}[k]$, $\psi^{2r}_{[1],1}[k]$, which originate from the discrete splines of order 2, 4 and 8, respectively, and their spectra.

Best basis

A set of discrete spline wavelet packets has the same tree structure as the set of polynomial spline wavelet packets. Like in the polynomial case, a variety of the bases generated by shifts of wavelet packets is available. A combination of wavelet packets can generate a basis for the spline space ${}^{2r}\mathscr{S}$ if their DFT spectra form a one-fold cover of the whole spectral domain $n = -N/2, ...N/2 - 1$. A basis that is optimal, according to some chosen cost function, can be designed using the Best Basis methodology, which is described in Sect. 8.4.4. Once the wavelet packet transform of a signal is implemented and the entropies in each block of the transform coefficients are computed by the MATLAB function `D_WP_analP.m`, the list of the wavelet packets, whose shifts constitute the best basis tree is provided by the function `WP_BB_Listree1D_P.m`. The MATLAB function `D_List_synthP.m` restores the signal $\mathbf{x}$ from the basis defined by a list Λ, using the relation Eq. (14.36).

14.4 2D Wavelet Packets

As in Sect. 13.2, we assume that the two-dimensional discrete splines have the same periodicity N in both vertical and horizontal directions. The spaces of such splines are denoted as ${}^{2r,2r}\mathscr{S}_{[m]}$. We keep the notations

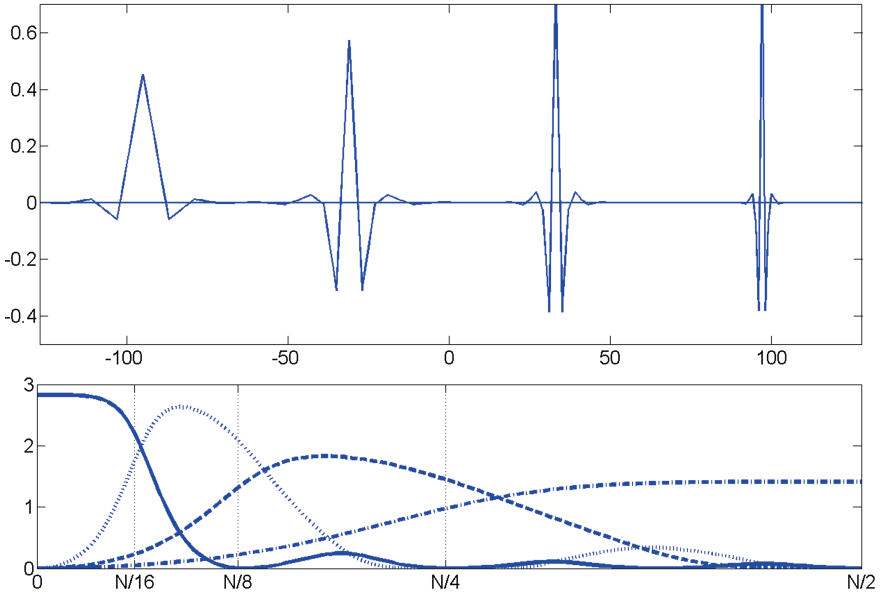


Fig. 14.12 Discrete spline wavelets of order 2 and the right half-band of their SHA spectra. *Upper frame left* to *right* $\psi^2_{[3],0}$, $\psi^2_{[3],1}$, $\psi^2_{[2],1}$, $\psi^2_{[1],1}$. *Bottom frame* $\tau_{[3],0}[\nu]$, (*solid line*) $\tau_{[3],1}[\nu]$ (*dotted*), $\tau_{[2],1}[\nu]$ (*dashed*), $\tau_{[1],1}[\nu]$ (*dash-dotted*)

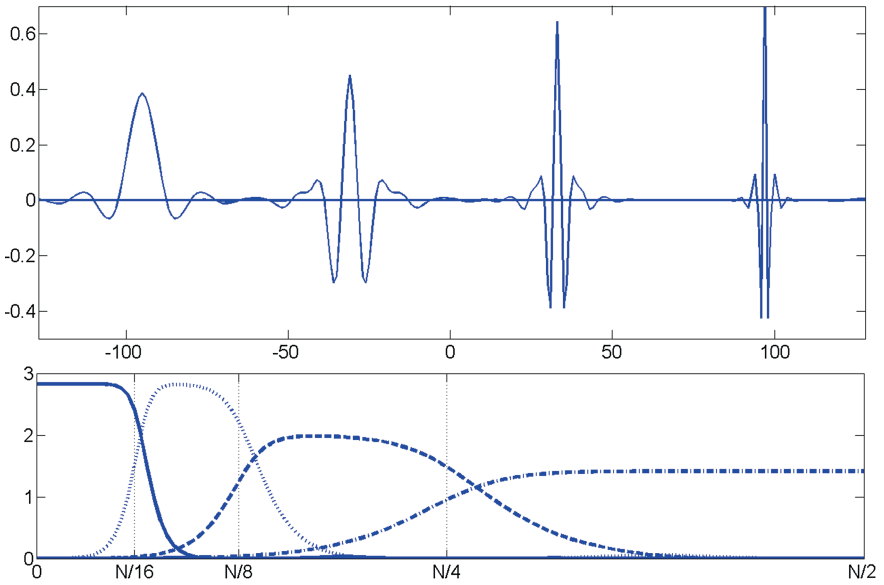


Fig. 14.13 Discrete spline wavelets of order 4 and the right half-band of their SHA spectra. *Upper frame left* to *right* $\psi^4_{[3],0}$, $\psi^4_{[3],1}$, $\psi^4_{[2],1}$, $\psi^4_{[1],1}$. *Bottom frame* $\tau_{[3],0}[\nu]$, (*solid line*) $\tau_{[3],1}[\nu]$ (*dotted*), $\tau_{[2],1}[\nu]$ (*dashed*), $\tau_{[1],1}[\nu]$ (*dash-dotted*)

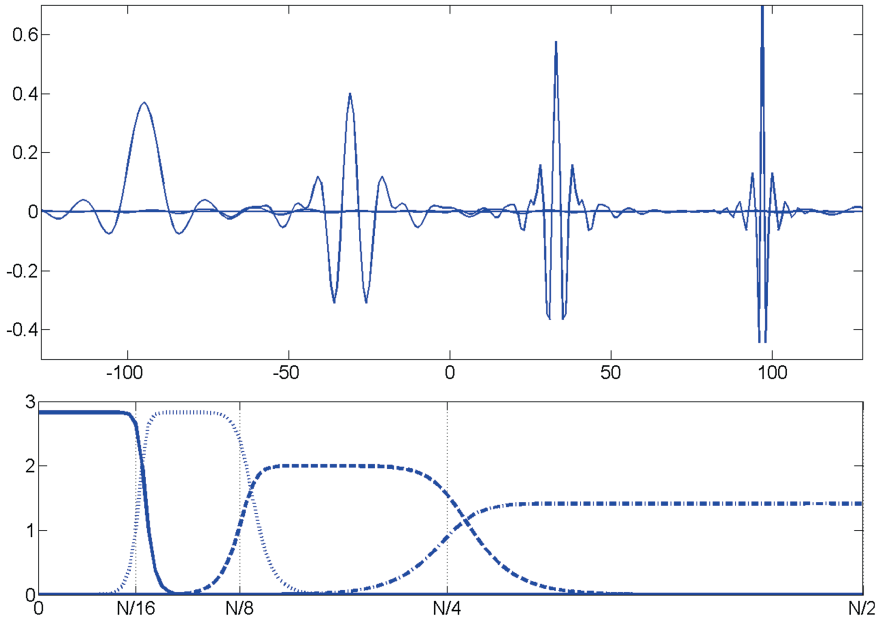


Fig. 14.14 Discrete spline wavelets of order 8 and the right half-band of their SHA spectra. *Upper frame left* to *right* $\psi^8_{[3],0}$, $\psi^8_{[3],1}$, $\psi^8_{[2],1}$, $\psi^8_{[1],1}$. *Bottom frame* $\tau_{[3],0}[\nu]$, (*solid line*) $\tau_{[3],1}[\nu]$ (*dotted*), $\tau_{[2],1}[\nu]$ (*dashed*), $\tau_{[1],1}[\nu]$ (*dash-dotted*)

$$\gamma^{2r}_{[m]}(\kappa,\nu) = \left\{\tilde{\gamma}^{2r}_{[m]}(\kappa,\nu)[k,n] \stackrel{\text{def}}{=} \gamma^{2r}_{[m]}(\kappa)[k]\,\tilde{\gamma}^{2r}_{[m]}(\nu)[n]\right\},$$

$$U^{2r}_{[m]}[\kappa,\nu] \stackrel{\text{def}}{=} U^{2r}_{[m]}[\kappa]\,U^{2r}_{[m]}[\nu], \quad \kappa,\nu = -N_m/2,\ldots,N_m/2-1,\ N_m = N/2^m,$$

$$\gamma^{2r}_{[m],l,\tilde{l}}(\kappa,\nu) = \left\{\tilde{\gamma}^{2r}_{[m],l,\tilde{l}}(\kappa,\nu)[k,n] \stackrel{\text{def}}{=} \gamma^{2r}_{[m],l}(\kappa)[k]\,\tilde{\gamma}^{2r}_{[m],\tilde{l}}(\nu)[n]\right\}, \quad l,\tilde{l} = 0,\ldots,2^m-1.$$

14.4.1 2D Discrete Spline Wavelet Packets Bases

The discrete splines $\gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n]$ with the same level index m are mutually orthogonal and their norms are

$$\left\|\gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu]\right\| = \sqrt{\sum_{k,n=0}^{N-1}\left|\gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n]\right|^2} = 1.$$

The 2D discrete spline spaces ${}^{2r}\mathscr{W}_{[m],l,\tilde{l}} \subset \Pi[N,N]$ are defined as the linear spans of the splines $\gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n]$:

$$s_{[m],l,\tilde{l}}[k,n] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \sigma_{[m],l,\tilde{l}}[\kappa,\nu] \gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n] \in {}^{2r}\mathscr{W}_{[m],l,\tilde{l}}. \quad (14.39)$$

The splits of the 1D signal space $\Pi[N]$, which are illustrated in diagram in Fig 14.1, generate the splits of the 2D signal space $\Pi[N,N]$ into mutually orthogonal subspaces:

$$\begin{aligned} {}^{2r}\mathscr{W}_{[m-1],l,\tilde{l}} &= {}^{2r}\mathscr{W}_{[m],2l,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{[m],2l+1,2\tilde{l}} \bigoplus {}^{2r}\mathscr{W}_{[m],2l,2\tilde{l}+1} \bigoplus {}^{2r}\mathscr{W}_{[m],2l+1,2\tilde{l}+1} \\ \Longrightarrow \Pi[N,N] &= \mathscr{W}_{[0],0,0} = {}^{2r}\mathscr{W}_{[1],0,0} \bigoplus {}^{2r}\mathscr{W}_{[1],0,1} \bigoplus {}^{2r}\mathscr{W}_{[1],1,0} \bigoplus {}^{2r}\mathscr{W}_{[1],1,1} \\ &\quad \cdots\cdots\cdots \\ &= \bigoplus_{l,\tilde{l}=0}^{2^m-1} {}^{2r}\mathscr{W}_{[m],l,\tilde{l}}, \quad m = 2, ..., M. \end{aligned}$$

As an alternative to the orthonormal bases $\left\{\gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n]\right\}$, $\kappa,\nu = -N_m/2, ..., N_m/2-1$, of the subspaces ${}^{2r}\mathscr{W}_{[m],l,\tilde{l}} \subset \mathscr{W}_{[0],0,0} = \Pi[N.N]$, there exist orthonormal bases that consist of shifts of the 2D wavelet packets

$$\mathbf{s}_{[m],l,\tilde{l}}[k,n] = \sum_{\lambda,\mu=0}^{N_m-1} q_{[m],l,\tilde{l}}[\lambda,\mu] \psi^{2r}_{[m],l,\tilde{l}}[k-2^m\lambda, n-2^m\mu), \quad (14.40)$$

$$\psi^{2r}_{[m],l,\tilde{l}}[k,n] \stackrel{\text{def}}{=} \psi^{2r}_{[m],l}[k]\, \psi^{2r}_{[m],\tilde{l}}[n] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \gamma^{2r}_{[m],l,\tilde{l}}[\kappa,\nu][k,n],$$

$$q_{[m],l,\tilde{l}}[\lambda,\mu] = \frac{1}{N_m} \sum_{\kappa,\nu=-N_m/2}^{N_m/2-1} \omega^{2^m(\mu\nu+\lambda\kappa)}\, \sigma_{[m],l,\tilde{l}}[\kappa,\nu]. \quad (14.41)$$

The discrete spline $\psi^{2r}_{[m],0,0}[k,n] \in {}^{2r,2r}\mathscr{W}_{[m],0,0}$ is the 2D periodic father wavelet, while the triple $\psi^{2r}_{[m],1,0}[k,n]$, $\psi^{2r}_{[m],0,1}[k,n]$ and $\psi^{2r}_{[m],1,1}[k,n]$ are the 2D periodic wavelets from level m.

Two-dimensional wavelets and wavelet packets, which belong to the signal space $\Pi[N,N]$, can be represented via the IDFT:

$$\psi^{2r}_{[m],l,\tilde{l}}[k,n] = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \tau_{[m],l,\tilde{l}}[\kappa,\nu]\, \omega^{(\kappa\kappa+n\nu)}. \quad (14.42)$$

Their DFT spectra $\left\{\tau_{[m],l,\tilde{l}}[\kappa,\nu] = \tau_{[m],l}[\kappa]\tau_{[m],\tilde{l}}[\nu]\right\}$ are the tensor products of the DFT spectra of the 1D wavelet packets.

14.4.2 2D Wavelet Packets Transforms

Assume that a signal $\mathbf{x} = \{x[k,n]\} \in \Pi[N,N] = {}^{\mathscr{W}}_{[0],0,\tilde{0}}$ and $\sigma_{[0],0,\tilde{0}}[\kappa,\nu] \stackrel{\text{def}}{=} \hat{x}[\kappa,\nu]$ is its 2D DFT spectrum. Then, the coordinates $\left\{\sigma_{[m],l,\tilde{l}}[\kappa,\nu]\right\}, m = 1, ..., M$, of its projections $\mathbf{s}_{[m],l,\tilde{l}}[k,n]$ (Eqs. 14.39) onto the subspaces and ${}^{2r}\mathscr{W}_{[m],l,\tilde{l}}$ are derived by application of the 2D wavelet packet transforms to the coordinates $\left\{\sigma_{[0],0,\tilde{0}}[\kappa,\nu]\right\}$, $\kappa,\nu = -N/2, ..., N/2-1$. The 2D transform can be implemented by iterated application of the 1D transform subsequently to the rows and columns of the array $\left\{\sigma_{[m]}[\kappa,\nu]\right\}$. If the projections coordinates $\left\{\sigma_{[m],l,\tilde{l}}[\kappa,\nu]\right\}$ at some level m are available then the DFT $\hat{x}[\kappa,\nu] = \left\{\sigma_{[0],0,0}[\kappa,\nu]\right\}$, $\kappa,\nu = -N/2, ..., N/2-1$ are restored by application of the 2D inverse wavelet packet transforms to the combined coefficients array

$$\mathbf{C}_m = \begin{pmatrix} \left\{\sigma_{[m],0,0}[\kappa,\nu]\right\} & \left\{\sigma_{[m],0,1}[\kappa,\nu]\right\} & ... & \left\{\sigma_{[m],0,M}[\kappa,\nu]\right\} \\ \left\{\sigma_{[m],1,0}[\kappa,\nu]\right\} & \left\{\sigma_{[m],1,1}[\kappa,\nu]\right\} & ... & \left\{\sigma_{[m],1,M}[\kappa,\nu]\right\} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \left\{\sigma_{[m],l,\tilde{l}}[\kappa,\nu]\right\} & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \left\{\sigma_{[m],M,0}[\kappa,\nu]\right\} & \left\{\sigma_{[m],M,1}[\kappa,\nu]\right\} & ... & \left\{\sigma_{[m],M,M}[\kappa,\nu]\right\} \end{pmatrix},$$

where $M \stackrel{\text{def}}{=} 2^m - 1$ and $\kappa,\nu = -N_m/2, ..., N_m/2-1$. Note that the size of the array $\mathbf{C}_m$ is $N \times N$.

Assume that a spline $\mathbf{s}_{[m],l,\tilde{l}}[k,n] \in {}^{2r}\mathscr{W}_{[m],l,\tilde{l}}$ is expanded as in Eqs. (14.39). Its DFT spectrum is the coefficients set $\left\{\varsigma_{[m],1,\tilde{1}}[\kappa,\nu]\right\}$, $\kappa,\nu = -N/2, ..., N/2-1$, of its representation in the initial space ${}^{2r,2r}\mathscr{S}$

$$s_{[m],l,\tilde{l}}[k,n] = \frac{1}{N^2} \sum_{\kappa,\nu=-N/2}^{N/2-1} \varsigma_{[m],l,\tilde{l}}[\kappa,\nu]\, \omega^{(k\kappa+b\nu)}.$$

The DFT spectrum can be derived by application of the 2D inverse wavelet packet transforms to the coefficients array $\mathbf{C}_m$, where all the components except for $\left\{\sigma_{[m],l,\tilde{l}}[\kappa,\nu]\right\}$ are put to zero.

14.4.2.1 2D Transform Matrices

Similarly to 1D transforms, direct and inverse 2D discrete spline wavelet packet transforms can be explicitly expressed in a matrix form. The transforms are imple-

mented along the scheme, which is identical to the scheme for the 2D continuous wavelet packet transforms described in Sect. 8.5.2.

Expanding the matrices $\mathbf{A}_{[m-1]}[\nu]$ and $\mathbf{A}^{\dagger}_{[m-1]} = \mathbf{A}^{-1}_{[m-1]}$ defined in Eqs. (14.12) and (14.13), respectively, to the 2D case, we define

$$\mathbf{A}_{[m-1]}[\kappa, \nu] \stackrel{\text{def}}{=} \mathbf{A}_{[m-1]}[\kappa] \bigotimes \mathbf{A}_{[m-1]}[\nu] \tag{14.43}$$

$$= \begin{pmatrix} s_{[m-1]}[\kappa]\, s_{[m-1]}[\nu] & s_{[m-1]}[\kappa]\, s_{[m-1]}[\bar{\nu}] & s_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\nu] & s_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\bar{\nu}] \\ s_{[m-1]}[\kappa]\, d_{[m-1]}[\nu] & s_{[m-1]}[\kappa]\, d_{[m-1]}[\bar{\nu}] & s_{[m-1]}[\bar{\kappa}]\, d_{[m-1]}[\nu] & s_{[m-1]}[\bar{\kappa}]\, d_{[m-1]}[\bar{\nu}] \\ d_{[m-1]}[\kappa]\, s_{[m-1]}[\nu] & d_{[m-1]}[\kappa]\, s_{[m-1]}[\bar{\nu}] & d_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\nu] & d_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\bar{\nu}] \\ d_{[m-1]}[\kappa]\, d_{[m-1]}[\nu] & d_{[m-1]}[\kappa]\, d_{[m-1]}[\bar{\nu}] & d_{[m-1]}[\bar{\kappa}]\, d_{[m-1]}[\nu] & d_{[m-1]}[\bar{\kappa}]\, d_{[m-1]}[\bar{\nu}] \end{pmatrix}$$

$$s_{[m-1]}(\nu) \stackrel{\text{def}}{=} \sqrt{\frac{U^{4r}_{[m-1]}[\nu]}{2\, U^{4r}_{[m]}[\nu]}}\, \cos^{2r} \frac{\pi\, \nu}{N_{m-1}}, \quad d_{m-1}(\nu) \stackrel{\text{def}}{=} \omega^{2^{m-1}\nu}\, s_{m-1}(\nu + N_m).$$

$$\mathbf{A}^{-1}_{[m-1]}[\kappa, \nu] = \mathbf{A}^{-1}_{[m-1]}[\kappa] \bigotimes \mathbf{A}^{-1}_{[m-1]}[\nu] = \mathbf{A}^{\dagger}_{[m-1]}[\kappa] \bigotimes \mathbf{A}^{\dagger}_{[m-1]}[\nu] = \mathbf{A}^{\dagger}_{[m-1]}[\kappa, \nu]$$

$$= \begin{pmatrix} s_{[m-1]}[\kappa]\, s_{[m-1]}[\nu] & s_{[m-1]}[\kappa]\, d^{*}_{[m-1]}[\nu] & d^{*}_{[m-1]}[\kappa]\, s_{[m-1]}[\nu] & d^{*}_{[m-1]}[\kappa]\, d^{*}_{[m-1]}[\nu] \\ s_{[m-1]}[\kappa]\, s_{[m-1]}[\bar{\nu}] & s_{[m-1]}[\kappa]\, d^{*}_{[m-1]}[\bar{\nu}] & d^{*}_{[m-1]}[\kappa]\, s_{[m-1]}[\bar{\nu}] & d^{*}_{[m-1]}[\kappa]\, d^{*}_{[m-1]}[\bar{\nu}] \\ s_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\nu] & s_{[m-1]}[\bar{\kappa}]\, d^{*}_{[m-1]}[\nu] & d^{*}_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\nu] & d^{*}_{[m-1]}[\bar{\kappa}]\, d^{*}_{[m-1]}[\nu] \\ s_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\bar{\nu}] & s_{[m-1]}[\bar{\kappa}]\, d^{*}_{[m-1]}[\bar{\nu}] & d^{*}_{[m-1]}[\bar{\kappa}]\, s_{[m-1]}[\bar{\nu}] & d^{*}_{[m-1]}[\bar{\kappa}]\, d^{*}_{[m-1]}[\bar{\nu}] \end{pmatrix}.$$

14.4.2.2 2D Discrete Spline Wavelet Transform

The $\bar{m}$ -level wavelet transform consists in decomposition of a signal $\mathbf{x} = \{x[k, n]\} \in \Pi[N, N]$ such that $\sigma_{[0],0,\tilde{0}}[\kappa, \nu] \stackrel{\text{def}}{=} \hat{x}[\kappa, \nu]$ into the orthogonal sum

$$\begin{aligned} x[k, n] = {} & \mathbf{s}_{[\bar{m}],0,0}[k, n] + \mathbf{s}_{[\bar{m}],1,0}[k, n] + \mathbf{s}_{[\bar{m}],0,1}[k, n] + \mathbf{s}_{[\bar{m}],1,1}[k, n] \\ & + \mathbf{s}_{[\bar{m}-1],1,0}[k, n] + \mathbf{s}_{[\bar{m}-1],0,1}[k, n] + \mathbf{s}_{[\bar{m}-1],1,1}[k, n] + \cdots \\ & + \mathbf{s}_{[1],1,0}[k, n] + \mathbf{s}_{[1],0,1}[k, n] + \mathbf{s}_{[1],1,1}[k, n]. \end{aligned} \tag{14.44}$$

The splines $\mathbf{s}_{[m],l,\tilde{l}}[k, n]$ are represented either via orthonormal bases of exponential splines as in Eq. (14.39), or via orthonormal bases of shifts of the wavelets $\psi^{2r}_{[m],l,\tilde{l}}[k, n]$, $l.\tilde{l} = 0, 1$, as in Eq. (14.40). The coordinates in two representations are linked by Eq. (14.41).

The coordinates $\sigma_{[m],l,\tilde{l}}[k, n]$, $l.\tilde{l} = 0, 1,$, of the signal $\mathbf{x} \in \Pi[N, N]$ projections onto the subspaces ${}^{2r}\mathscr{W}_{[m],l,\tilde{l}}[k, n]$, $l.\tilde{l} = 0, 1,$, respectively, are derived from the coordinates $\sigma_{[m-1],0,0}[\kappa, \nu]$ of the spline $\mathbf{s}_{[m-1],0,0}[k, n] \in {}^{2r,2r}\mathscr{W}_{[m-1],0,0}$ in a way similar to Eq. (14.24):

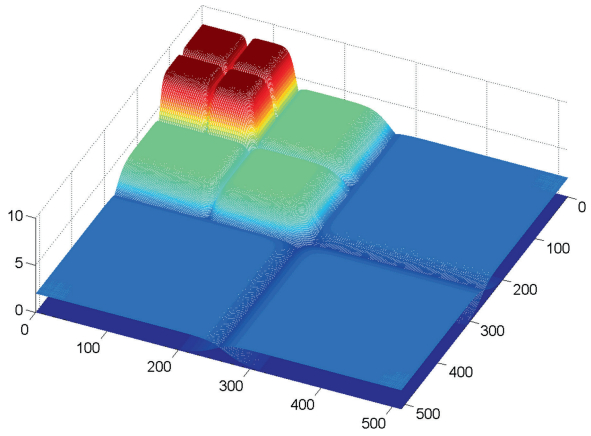


Fig. 14.15 Magnitudes of the 2D DFT of discrete splines wavelets of order 8 from three decomposition levels: $\psi^{8}_{[3],0,0}[k,n]$, $\psi^{8}_{[m],1,0}[k,n]$, $\psi^{8}_{[m],0,1}[k,n]$, $\psi^{8}_{[m],1,1}[k,n]$, $m=1,2,3$

$$\begin{pmatrix} \sigma_{[m],0,0}[\kappa,\nu] \\ \sigma_{[m],1,0}[\kappa,\nu] \\ \sigma_{[m],0,1}[\kappa,\nu] \\ \sigma_{[m],1,1}[\kappa,\nu] \end{pmatrix} = \frac{1}{2}\mathbf{A}_{[m-1]}[\kappa,\nu]\cdot \begin{pmatrix} \sigma_{[m-1],0,0}[\kappa,\nu] \\ \sigma_{[m-1],0,0}[\kappa,\bar{\nu}] \\ \sigma_{[m-1],0,0}[\bar{\kappa},\nu] \\ \sigma_{[m-1],0,0}[\bar{\kappa},\bar{\nu}] \end{pmatrix}, \quad \kappa,\nu=-N_m/2,...,N_m/2-1, \tag{14.45}$$

and $\bar{\kappa} \stackrel{\text{def}}{=} \kappa + N_m/2$ and $\bar{\nu} \stackrel{\text{def}}{=} \nu + N_m/2$. Having the coefficients $\sigma_{[m],l,\tilde{l}}[\kappa,\nu]$, $m=1,...M$, $l,\tilde{l}=0,1$, the wavelet coefficients $q_{[m],l,\tilde{l}}[\lambda,\mu]$ are calculated using the IDFT, as presented in Eq. (14.41).

The reconstruction of $x[k,n]$ is implemented iteratively, starting from the bottom level $\bar{m}$ similarly to Eq. (14.25):

$$\begin{pmatrix} \sigma_{[m-1],0,0}[\kappa,\nu] \\ \sigma_{[m-1],0,0}[\kappa,\bar{\nu}] \\ \sigma_{[m-1],0,0}[\bar{\kappa},\nu] \\ \sigma_{[m-1],0,0}[\bar{\kappa},\bar{\nu}] \end{pmatrix} = 2\mathbf{A}^{\dagger}_{[m-1]}[\kappa,\nu]\cdot \begin{pmatrix} \sigma_{[m],0,0}[\kappa,\nu] \\ \sigma_{[m],1,0}[\kappa,\nu] \\ \sigma_{[m],0,1}[\kappa,\nu] \\ \sigma_{[m],1,1}[\kappa,\nu] \end{pmatrix}, \tag{14.46}$$

where $\kappa,\nu=-N_m/2,...,N_m/2-1$. Application of the transforms Eq. (14.46) can produce the DFT spectra of the projections splines and, in particular, the DFT spectra of the wavelets $\psi^{2r}_{[m],l,\tilde{l}}[k,n]$, $(l,\tilde{l})=0,1$. In a sense, these DFT spectra provide a logarithmic-wise partition of the DFT spectral square $\kappa,\nu=-N/2,...,N/2-1$.

The 2D discrete spline wavelet transform and its inverse are implemented by the MATLAB functions `D_wav_anal_2dP.m` and `D_wav_syntl_2dP.m`, respectively.

Figure 14.15, which is produced by the MATLAB code `D_mesh_wp_listP.m`, displays the magnitudes of the DFT of discrete spline wavelets of order 8 from three decomposition levels. For clarity, we display only the first quadrant $\kappa,\nu=$

$0, ..., N/2-1$ of the spectral square. The magnitudes of the DFT in the other quadrants mirror those in the first quadrant.

14.4.2.3 2D Wavelet Packet Transform

The $\bar{m}$ -level wavelet packet transform of order $2r$ of a signal $\mathbf{x} \in \Pi[N, N]$ results in decomposition of $\mathbf{x}$ into $\bar{m}$ orthogonal sums

$$\mathbf{x} = \mathbf{s}_{[1],0,0} + \mathbf{s}_{[1],1,0} + \mathbf{s}_{[1],0,1} + \mathbf{s}_{[1],1,1} = \sum_{l,\tilde{l}=0}^{3} \mathbf{s}_{[2],l,\tilde{l}} = ... = \sum_{l,\tilde{l}=0}^{2^{\bar{m}}-1} \mathbf{s}_{[\bar{m}],l,\tilde{l}}. \quad (14.47)$$

The splines $\mathbf{s}_{[m],l,\tilde{l}} \in {}^{2r}\mathscr{W}_{[m],l,\tilde{l}}$ are represented either via orthonormal bases of exponential splines as in Eq. (14.39) or via orthonormal bases of shifts of the wavelet packets $\psi^{2r}_{[m],l,\tilde{l}}[k, n]$ as in Eq. (14.40). The coordinates in the two representations are linked by Eq. (14.41).

The coordinates $\sigma_{[m],2l,2\tilde{l}}[\kappa, \nu]$, $\sigma_{[m],2l+1,2\tilde{l}}[\kappa, \nu]$, $\sigma_{[m],2l,2\tilde{l}+1}[\kappa, \nu]$ and $\sigma_{[m],2l+1,2\tilde{l}+1}[\kappa, \nu]$ of the splines $\mathbf{s}_{[m],2l,2\tilde{l}}$, $\mathbf{s}_{[m],2l+1,2\tilde{l}}$, $\mathbf{s}_{[m],2l,2\tilde{l}+1}$ and $\mathbf{s}_{[m],2l+1,2\tilde{l}+1}$, respectively, are derived from the coordinates $\sigma_{[m-1],l,\tilde{l}}[\kappa, \nu]$ of the spline $\mathbf{s}_{[m-1],l,\tilde{l}} \in {}^{2r}\mathscr{W}_{m-1,l,\tilde{l}}$ using the matrices $\mathbf{A}_{[m-1]}[\kappa, \nu]$ defined in Eq. (14.43).

$$\begin{pmatrix} \sigma_{[m],\mathbf{l}_1}[\kappa, \nu] \\ \sigma_{[m],\mathbf{l}_2}[\kappa, \nu] \\ \sigma_{[m],\mathbf{l}_3}[\kappa, \nu] \\ \sigma_{[m],\mathbf{l}_4}[\kappa, \nu] \end{pmatrix} = \frac{1}{2}\mathbf{A}_{[m-1]}[\kappa, \nu] \cdot \begin{pmatrix} \sigma_{[m-1],l,\tilde{l}}[\kappa, \nu] \\ \sigma_{[m-1],l,\tilde{l}}[\kappa, \bar{\nu}] \\ \sigma_{[m-1],l,\tilde{l}}[\bar{\kappa}, \nu] \\ \sigma_{[m-1],l,\tilde{l}}[\bar{\kappa}, \bar{\nu}] \end{pmatrix}, \quad (14.48)$$

where $\kappa, \nu = -N_m/2, ..., N_m/2-1$, $\bar{\kappa} \stackrel{\text{def}}{=} \kappa + N_m/2$, $\bar{\nu} \stackrel{\text{def}}{=} \nu + N_m/2$ and

$$(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4) \quad (14.49)$$
$$= \begin{cases} ((2l, 2\tilde{l}), (2l+1, 2\tilde{l}), (2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}+1)), \text{ if } l \text{ even}, \tilde{l} \text{ even}; \\ ((2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}), (2l+1, 2\tilde{l})), \text{ if } l \text{ even}, \tilde{l} \text{ odd}; \\ ((2l, 2\tilde{l}+1), (2l, 2\tilde{l}), (2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}+1)), \text{ if } l \text{ odd}, \tilde{l} \text{ even}; \\ ((2l+1, 2\tilde{l}+1), (2l, 2\tilde{l}+1), (2l+1, 2\tilde{l}), (2l, 2\tilde{l})), \text{ if } l \text{ odd}, \tilde{l} \text{ odd}. \end{cases}$$

The 2D discrete spline wavelet packet transform, of an array $\mathbf{x}$, which produces its coordinates in the wavelet packet bases, is implemented by the MATLAB function `D_WP_anal_2DP.m`.

The reconstruction of $x[k, n]$ is implemented iteratively starting from any level m similarly to Eq. (14.46):

$$\begin{pmatrix} \sigma_{[m-1],l,\tilde{l}}[\kappa,\nu] \\ \sigma_{[m-1],l,\tilde{l}}[\kappa,\bar{\nu}] \\ \sigma_{[m-1],l,\tilde{l}}[\bar{\kappa},\nu] \\ \sigma_{[m-1],l,\tilde{l}}[\bar{\kappa},\bar{\nu}] \end{pmatrix} = 2\mathbf{A}^{\dagger}_{m-1}[\kappa,\nu] \cdot \begin{pmatrix} \sigma_{[m],\mathbf{l}_1}[\kappa,\nu] \\ \sigma_{[m],\mathbf{l}_2}[\kappa,\nu] \\ \sigma_{[m],\mathbf{l}_3}[\kappa,\nu] \\ \sigma_{[m],\mathbf{l}_4}[\kappa,\nu] \end{pmatrix}, \tag{14.50}$$

where $\kappa, \nu = -N_m/2, ..., N_m/2-1$ and the multiindices $(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4)$ are defined by Eq. (14.49). Application of the transforms Eq. (14.50) can produce the DFT spectra of the projections splines $\mathbf{s}_{[m],l,\tilde{l}}[k,n] \in {}^{2r}\mathscr{W}_{[m],l,\tilde{l}}$ and, in particular, the DFT spectra of the wavelet packets $\psi^{2r}_{[m],l,\tilde{l}}[k,n]$. These DFT spectra provide a variety of partitions of the DFT spectral square $\kappa, \nu = -N/2, ..., N/2-1$. Consequently, the shifts of the wavelet packets provide a variety of the orthonormal bases for the spline space ${}^{2r,2r}\mathscr{S}$. As in the 1D setting, the following special cases are possible:

"Horizontal" bases

These are bases consisting of shifts of all the wavelet packets from the same level m, where m can be one of the integers $1, ..., \bar{m}$:

$$\mathbf{B}^{2h}_{m} \stackrel{\text{def}}{=} \bigcup_{l,\tilde{l}=0}^{2^m-1} \left\{ \psi^{2r}_{[m],1,\tilde{l}}[k-2^m\lambda, n-2^m\mu] \right\}, \tag{14.51}$$

where $\lambda, \mu = 0, ..., N_m-1$, $N_m = N/2^m$. In this case, the DFT spectrum of a signal $\mathbf{x}$ is split into 2^{2m} regions of the same size.

Reconstruction of the array $\mathbf{x}$ from the coefficients in a "horizontal" basis is implemented by the MATLAB function `D_hor_WP_synth_2DP.m`.

Wavelet bases

$$\begin{aligned} \mathbf{B}^{2w}_{m} \stackrel{\text{def}}{=} & \left\{ \psi^{2r}_{[m],0,0}[k-2^m\lambda, n-2^m\mu] \right\}_{\lambda,\mu=0}^{N_m-1} \bigcup \left\{ \psi^{2r}_{[m],1,0}[k-2^m\lambda, n-2^m\mu] \right\}_{\lambda,\mu=0}^{N_m-1} \\ & \bigcup \left\{ \psi^{2r}_{[m],0,1}[k-2^m\lambda, n-2^m\mu] \right\}_{\lambda,\mu=0}^{N_m-1} \bigcup \left\{ \psi^{2r}_{[m],0,0}[k-2^m\lambda, n-2^m\mu] \right\}_{\lambda,\mu=0}^{N_m-1} \bigcup \\ & ... \bigcup \left\{ \psi^{2r}_{[1],0,1}[k-2\lambda, n-2\mu] \right\}_{\lambda,\mu=0}^{N/2-1} \bigcup \left\{ \psi^{2r}_{[1],1,0}[k-2\lambda, n-2\mu] \right\}_{\lambda,\mu=0}^{N/2-1} \\ & \bigcup \left\{ \psi^{2r}_{[1],1,1}[k-2\lambda, n-2\mu] \right\}_{\lambda,\mu=0}^{N/2-1}. \end{aligned}$$

Besides these two options, any combination of wavelet packets such that their DFT spectra form a one-fold cover of the whole spectral domain $\kappa, \nu = -N/2, \ldots N/2-1$, generate an orthonormal basis for the signal space $\Pi[N,N]$.

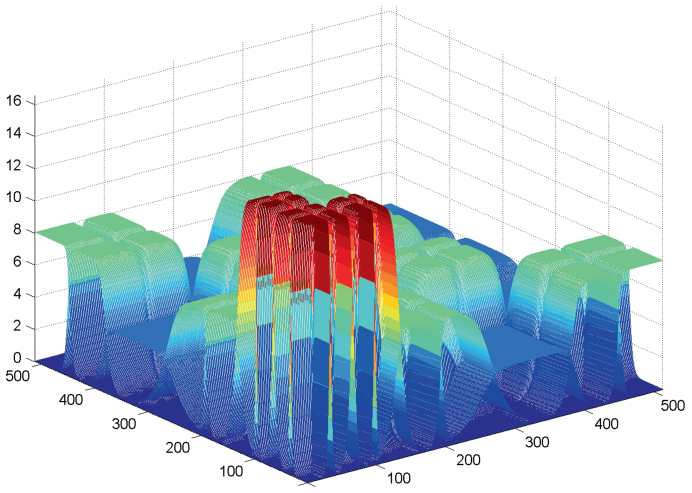


Fig. 14.16 Magnitudes of the DFT discrete spline wavelet packets $\psi^8_{[m],l,\tilde{l}}[k,n]$, of eighth order from the second to fourth decomposition levels, whose shifts constitute the best basis for the "Lena" image

Best basis

Among all the available wavelet packet bases, a basis that is optimal with respect to a cost function, can be selected. It is achieved via the Best Basis methodology , which is described in Sect. 8.4.4. The 2D case is outlined in Sect. 8.5.3. The list of wavelet packets, whose shifts constitute a Best Basis is compiled by the MATLAB function `WP_BB_Listree2_D_P.m`. The MATLAB function `D_List_synthP_2D.m` restores the array **x** from the basis defined by a list Λ.

Figure 14.16, which was produced by the MATLAB code `D_mesh_wp_listP.m`, displays the magnitudes of the DFT of the discrete spline wavelet packets $\psi^8_{[m],l,\tilde{l}}[k,n]$, of eighth order from the second to fourth decomposition levels, whose shifts constitute the Best Basis for the "Lena" image defined by a 512×512 array of pixels. For clarity, we display only the first quadrant $\kappa, \nu = 0, ..., N/2 - 1$ of the spectral square. The DFT magnitudes in the other quadrants mirror those in the first quadrant.

Remarks on applications of discrete spline wavelet packets

The potential field of applications of the discrete spline wavelet packets is the same as the field of applications of the continuous wavelet packets. The field consists of, for example, feature extraction of classes of signals as in Chap. 12, deconvolution or denoising of digitally defined images and signals by block-based algorithms or by the regularized matching pursuit (RMP). The schemes for implementation of the algorithms are exactly the same as the respective schemes using continuous spline wavelet packets. The results of application of the discrete spline wavelet packets are very close to the results of application of the continuous spline wavelet packets. We briefly outline the RMP algorithm with the discrete spline wavelet packets dictionaries.

14.5 Deconvolution and Denoising by Regularized Matching Pursuit

The goal of the algorithm is restoration of a signal, which passed through a low-pass filter, the output from which has been affected by white noise. To be specific, we should restore a signal $\mathbf{f} = \{f[k]\} \in \pi[N]$ from the data $\mathbf{z} = \{z[k] = g[k] + e_k\}$, $k = 0, ..., N-1$, where

$$\mathbf{g} \stackrel{\text{def}}{=} \mathbf{h} \circledast \mathbf{f} \Longleftrightarrow g[k] = \sum_{l=0}^{N-1} h[k-l]\, f[l], \tag{14.52}$$

$\mathbf{h}$ is a low-pass p-filter and $\mathbf{e} = \{e_k\}$ is a zero-mean i.i.d. noise. Special cases include:

- A pure deconvolution when noise does not present.
- Denoising when low-pass filtering is not applied.

14.5.1 Outline of the RMP for Deconvolution

As in continuous setting, we utilize two dictionaries: one for testing the data $\mathbf{z}$ and the other for approximation of the sought-after signal $\mathbf{f}$.

Denote $\psi^{2r}_{[m],l,\lambda}[k] \stackrel{\text{def}}{=} \psi^{2r}_{[m],l}[k+\lambda]$, $\lambda = 0, ..., N-1$, where $\psi^{2r}_{[m],l}[k]$ is the discrete spline wavelet packet of order $2r$ from the space ${}^{2r}\mathscr{W}_{[m],l}$. Denote

$$\theta_{[m],l}[k] \stackrel{\text{def}}{=} \mathbf{h} \circledast \psi^{2r}_{[m],l}[k], \quad \theta_{[m],l,\lambda}[k] \stackrel{\text{def}}{=} \theta_{[m],l}[k+\lambda], \quad \lambda = 0, ..., N-1. \tag{14.53}$$

14.5.1.1 General Scheme

The algorithm is implemented via the following steps:

1. *Construction of two dictionaries:*

   Approximation dictionary $\mathbf{D}_A = \left\{\psi^{2r}_{[m],l,\lambda}[k]\right\}$, $\lambda = 0, ..., N-1$ consisting of one-sample circular shifts of discrete spline wavelet packets of different resolution spans.

   Testing dictionary $\mathbf{D}_T = \left\{\theta_{[m],l,\lambda}\right\}$ consisting of circular convolutions of shifts of discrete spline wavelet packets with the signal $\mathbf{h}$.

2. *Modeling the error vector* $\mathbf{e}$, that is a construction of a vector $\tilde{\mathbf{e}}$, whose characteristics are similar to the characteristics of $\mathbf{e}$.
3. *Calculation of oblique projections coefficients* $\alpha_{[m],l,\lambda}$ of $\mathbf{z}$ onto the normalized $\theta_{[m],l,\lambda}$. Setting a threshold T_s for the stopping rule.

4. If for some set $\breve{m}, \breve{l}, \breve{\lambda}$, the coefficient $|\alpha_{[\breve{m}],\breve{l},\breve{\lambda}}| < T_s$ then the signal $\theta_{[\breve{m}],\breve{l},\breve{\lambda}}$ is removed from the testing dictionary and the wavelet packet $\psi^{2r}_{[\breve{m}],\breve{l},\breve{\lambda}}[k]$ from the approximation dictionary.
5. The p-filter $\mathbf{h}$ is efficiently bandlimited. Therefore, the DFT spectra of some of the remaining wavelet packets $\psi^{2r}_{[\tilde{m}],\tilde{l}}$ actually do not overlap with the filter spectrum. In order to avoid instability of the algorithm, such wavelet packets and the corresponding $\theta_{[\tilde{m}],\tilde{l}}$, are discarded from the dictionaries.
6. Put $\mathbf{f}^0 \equiv 0$, $\mathbf{z}^0 = \mathbf{z}$.
7. Find *oblique* projections of $\mathbf{z}^0$ onto the normalized signals $\theta_{[m],l,\lambda}$:

$$\mathbf{z}^0_{[m],l,\lambda} = \alpha_{[m],l,\lambda} \frac{\theta_{[m],l,\lambda}}{\|\theta_{[m],l}\|}, \quad \alpha_{[m],l,\lambda} = \frac{\langle \theta_{[m],l,\lambda}, \mathbf{z}^0 \rangle}{\|\theta_{[m],l}\|} \left(1 - \frac{|\langle \theta_{[m],l,\lambda}, \tilde{\mathbf{e}} \rangle|}{|\langle \theta_{[m],l,\lambda}, \mathbf{z}^0 \rangle|} \right)_+,$$

   where $x_+ \stackrel{\text{def}}{=} (x + |x|)/2$.
8. Let $\bar{m}, \bar{l}, \bar{\lambda} = \arg\max_{[m],l,\lambda} |\alpha_{[m],l,\lambda}|$.
9. Denote $\Lambda^1 \stackrel{\text{def}}{=} \alpha_{\bar{m},\bar{l},\bar{\lambda}}/\|\theta_{[m],l}\|$, $\psi^1 \stackrel{\text{def}}{=} \psi^{2r}_{[\bar{m}],\bar{l},\bar{\lambda}}$ and $\Theta^1 \stackrel{\text{def}}{=} \theta_{[\bar{m}],\bar{l},\bar{\lambda}}$. Then, put $\mathbf{z}^1 = \mathbf{z}^0 - \Lambda^1 \Theta^1$, $\mathbf{f}^1 = \mathbf{f}^0 + \Lambda^1 \psi^1$.
10. The procedures starting from Item 7 are iterated, using $\mathbf{z}^1$ instead of $\mathbf{z}^0$, and so on.
11. The subsequent iterations of the algorithm extract pieces of coherent structure from the data signal $\mathbf{z}$. Thus, the noise-to-signal ratio tends to one, and the oblique projection coefficients tend to zero. When they become less than a predefined threshold T_s, the iterations are automatically aborted.
12. As a result, the signal $\mathbf{f}$ is approximated by a finite linear combination of discrete spline wavelet packets.

14.5.1.2 Dictionaries for RMP

For a successive approximation of $\mathbf{f} \in \Pi[N]$, we use the discrete spline wavelet packets $\psi^{2r}_{[m],l}$. Equation (14.32) implies

$$\psi^{2r}_{[m],l}[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\, \tau_{[m],l}[\nu] \Longrightarrow$$

$$\psi^{2r}_{[m],l,\lambda}[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\, \omega^{\lambda\nu}\, \tau_{[m],l}[\nu].$$

Since the available data $\mathbf{z}$ is the discrete convolution of $\mathbf{f}$ with the kernel $\mathbf{h}$, which was corrupted by noise, it is tested by the shifts of the convolved signals $\theta_{[m],l}[k] \stackrel{\text{def}}{=} \mathbf{h} \circledast \psi^{2r}_{[m],l}[k]$ defined in Eq. (14.53). Due to Eq. (1.25),

$$\theta^{2r}_{[m],l}[k] = \frac{1}{N}\sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\,\hat{h}[\nu]\,\tau_{[m],l}[\nu] \Longrightarrow$$

$$\theta^{2r}_{[m],l,\lambda}[k] = \frac{1}{N}\sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\,\omega^{\lambda\nu}\,\hat{h}[\nu]\,\tau_{[m],l}[\nu].$$

The dictionaries $\mathbf{D}_A$ and $\mathbf{D}_T$ are used for approximation of the solution and for testing the reminders, respectively:

$$\mathbf{D}_A = \left\{\psi^{2r}_{[m],l,\lambda}[k]\right\}_{\lambda=0}^{N-1}, \quad \psi^{2r}_{[m],l,\lambda}[k] = \frac{1}{N}\sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\,\omega^{\lambda\nu}\,\tau_{[m],l}[\nu],$$

$$\mathbf{D}_T = \left\{\theta_{[m],l,\lambda}\right\}_{\lambda=0}^{N-1}, \quad \theta^{2r}_{[m],l,\lambda}[k] = \frac{1}{N}\sum_{\nu=-N/2}^{N/2-1} \omega^{k\nu}\,\omega^{\lambda\nu}\,\hat{h}[\nu]\,\tau_{[m],l}[\nu],$$

where $\lambda, k = 0, ..., N-1$.

14.5.1.3 Modeling the Noise

Unknown real noise vector $\mathbf{e}$ is emulated by a vector $\tilde{\mathbf{e}}$, whose properties are similar to the properties of $\mathbf{e}$. The modeling scheme is described in Sect. 10.2.2.

14.5.1.4 Oblique Projection

As in continuous setting, orthogonal projection of the data onto dictionary elements can lead to the instability of the approximate solution. Therefore, oblique projection is implemented.

Repeating reasoning in Sect. 10.2.3, we derive an optimal oblique projection of the data sequence $\mathbf{z}$ onto a normalized element $\theta_{[m],l,\lambda}$ of the testing dictionary, using the noise model $\tilde{\mathbf{e}}$:

$$\mathbf{z}_{[m],l,\lambda} = \alpha_{[m],l,\lambda}\frac{\theta_{[m],l,\lambda}}{\|\theta_{[m],l}\|}, \quad \alpha_{[m],l,\lambda} = \frac{\langle\theta_{[m],l,\lambda},\mathbf{z}\rangle}{\|\theta_{[m],l}\|}\left(1 - \frac{|\langle\theta_{[m],l,\lambda},\tilde{\mathbf{e}}\rangle|}{|\langle\theta_{[m],l,\lambda},\mathbf{z}\rangle|}\right)_+. \tag{14.54}$$

14.5.1.5 Stopping Threshold and Reduction of Dictionaries

The scheme for setting a stopping threshold T_s and subsequent reduction of dictionaries is described in Sect. 10.2.4 for the polynomial spline wavelet packets.

Figure 10.4 illustrates that scheme. For the discrete splines case, the scheme is exactly the same.

Setting a threshold T_s provides a stopping rule for pursuit iterations.

14.5.1.6 Pursuit

The dictionary element $\theta_{[m],l,\lambda}$, which fits best the data array $\mathbf{z}^0$, is found from comparison of magnitudes of the oblique projection coefficients $\alpha_{[m],l,\lambda}$. Let $(\bar{m}, \bar{l}, \bar{\lambda}) = \arg\max_{[m],l,\lambda} \left|\alpha_{[m],l,\lambda}\right|$. Denote

$$\Lambda^1 \stackrel{\text{def}}{=} \frac{\alpha_{\bar{r},\bar{l},\bar{\lambda}}}{\|\theta_{[\bar{m}],\bar{l}}\|}, \quad \psi^1 \stackrel{\text{def}}{=} \psi^{2r}_{[\bar{m}],\bar{l},\bar{\lambda}}, \quad \Theta^1 \stackrel{\text{def}}{=} \theta_{\bar{r},\bar{l},\bar{\lambda}}.$$

Put

$$\mathbf{z}^1 = \mathbf{z}^0 - \Lambda^1 \Theta^1, \quad \mathbf{f}^1 = \mathbf{f}^0 + \Lambda^1 \psi^1.$$

Then, the procedure is iterated using $\mathbf{z}^1$ instead of $\mathbf{z}^0$.

It is seen from Eq. (14.54) that the oblique projection coefficients $\alpha_{[m],l,\lambda}$ depend on shares of the coherent structure and of noise in the projection of the remainder onto the normalized testing signal $\theta_{[m],l,\lambda}$. The bigger the noise share is, the smaller the coefficients. Subsequent pursuit steps extract pieces of coherent structure from the signal $\mathbf{z}$. Therefore, the noise share in the reminder increases and the oblique projection coefficients tend to zero. Thus, one after another, they appear below the threshold T_s. Consequently, one after another, signal $\theta_{[m],l}$ leaves the testing dictionary. At last, after some K iterations, the dictionary becomes empty thus terminating the iteration process. The approximate solution is

$$\tilde{\mathbf{f}} = \mathbf{f}^K = \sum_{k=1}^{K} \Lambda^k \psi^k. \tag{14.55}$$

14.5.2 Computational Scheme

14.5.2.1 Preprocessing

Dictionaries: Calculate the DFT of the kernel $\mathbf{h}$ $\hat{h}[\nu] = \sum_{k=0}^{N-1} \omega^{-k\nu} h[k]$.

Basing on a prior information on the solution $\mathbf{f}$ regularity, we choose the depth $\bar{m}$ and the order $2r$ of the discrete spline wavelet packets, which are used for the solution approximation, and calculate their DFT spectra, using Eq. (14.33):

$$\psi^{2r}_{[m],l,\lambda}[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \tau_{[m],l}[\nu]\, \omega^{(k+\lambda)\nu} \in {}^{2r}\mathscr{W}_{[m],l} \subset \Pi[N].$$

From the coefficients $\hat{h}[\nu]$ and $\tau_{[m],l}[\nu]$, we derive the elements

$$\theta_{[m],l,\lambda} = \left\{ \theta_{[m],l,\lambda}[k] = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \omega^{(k+\lambda)\nu}\, \hat{h}[\nu]\, \tau_{[m],l}[\nu] \right\}_{k,\lambda=0}^{N-1},$$

of the testing dictionary $\mathbf{D}_T$. For all $\lambda = 0, ..., N-1$, the norms are

$$\left\|\theta_{[m],l,\lambda}\right\| = \left\|\theta_{[m],l}\right\| = \sqrt{\sum_{i=0}^{N-1} \theta_{[m,]l}(i)^2} = \sqrt{\frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \left|\hat{h}[\nu]\, \tau_{[m],l}[\nu]\right|^2}.$$

Modeling the noise: Additive i.i.d. noise $\mathbf{e}$ is estimated as described in Sect. 10.2.2. The vector $\tilde{\mathbf{e}}$ is the estimation of $\mathbf{e}$ and $\left\{\hat{\tilde{e}}[n]\right\}$ is its DFT.

Setting threshold and reduction of dictionaries: Calculate the inner products $\left|\left\langle \theta_{[m],l,\lambda},\, \mathbf{z}\right\rangle\right|$ and $\left|\left\langle \theta_{[m],l,\lambda},\, \tilde{\mathbf{e}}\right\rangle\right|$, which are needed for the oblique projections (14.54), as

$$T_{[m],l,\lambda}(\mathbf{z}) \stackrel{\text{def}}{=} \left|\left\langle \theta_{[m],l,\lambda}, \mathbf{z}\right\rangle\right| = \frac{1}{N} \left| \sum_{\nu=-N/2}^{N/2-1} \omega^{\nu\lambda} \tau_{[m],l}[\nu]\, \hat{h}[\nu]\, \hat{z}[\nu]^* \right|, \quad (14.56)$$

$$e_{[m],l,\lambda} \stackrel{\text{def}}{=} \left|\left\langle \theta_{[m],l,\lambda}, \tilde{\mathbf{e}}\right\rangle\right| = \frac{1}{N} \left| \sum_{\nu=-N/2}^{N/2-1} \omega^{\nu\lambda} \tau_{[m],l}[\nu]\, \hat{h}; \widehat{\tilde{e}}[\nu]^* \right|.$$

Calculate the oblique projection coefficients

$$\acute{\alpha}_{[m],l,\lambda}(\mathbf{z}) = \frac{T_{[m],l,\lambda}(\mathbf{z})}{\left\|\theta_{[m],l}\right\|} \left(1 - \frac{e_{[m],l,\lambda}}{\left|T_{[m],l,\lambda}(\mathbf{z})\right|}\right)_+$$

and the values $A_{[m],l}(\mathbf{z}) \stackrel{\text{def}}{=} \max |\acute{\alpha}_{[m],l,\lambda}(\mathbf{z})|$ with respect to $\lambda = 0, ..., N-1$.

Set the threshold T_s and reduce dictionaries $\mathbf{D}_T$ and $\mathbf{D}_A$, as described in Sect. 10.2.4.

Initialization of the algorithm: The solution $\mathbf{f}$ of Eq. (14.52) is approximated by a sequence of signals

$$\mathbf{f}^j = \left\{ f^j[k] \right\}, \quad f^j[k] = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} \sigma^j[n]\, \omega^{kn} \in \Pi[N],$$

which are combinations of the discrete spline wavelet packets from the reduced dictionary $\tilde{\mathbf{D}}_A$.

Define the initial approximation and the initial remainder as

$$f^0[k] \equiv 0 \Longleftrightarrow \sigma^0[\nu] = 0 \ \forall \nu, \quad \mathbf{z}^0 = \mathbf{z} \Longleftrightarrow \hat{z}^0[\nu] = \hat{z}[\nu], \ \nu = -N/2, ..., N/2 - 1.$$

Then, $T_{[m],l,\lambda}(\mathbf{z^0}) = T_{[m],l,\lambda}(\mathbf{z}), \quad \acute{\alpha}_{[m],l,\lambda}(\mathbf{z}^0) = \acute{\alpha}_{[m],l,\lambda}(\mathbf{z})$.
Put $j = 0$.

14.5.2.2 Pursuit

1. Find $\bar{m}, \bar{l}, \bar{\lambda} = \arg\max_{[m],l,\lambda} \left|\alpha_{[m],l,\lambda}(\mathbf{z}^j)\right|$.
2. Denote $\Lambda^{j+1} \stackrel{\text{def}}{=} \alpha_{[\bar{m}],\bar{l},\bar{\lambda}} / \left\|\theta_{[\bar{m}],\bar{l}}\right\|$.
3. Put $\sigma^{j+1}[n] = \sigma^j[n] + \Lambda^{j+1}\, \tau_{\bar{r},\bar{l}}[\nu]\, \omega^{\bar{\lambda}\nu}, \ \nu = -N/2, ..., N/2 - 1$.
4. Put $\hat{\mathbf{z}}^{j+1}[\nu] = \hat{\mathbf{z}}^j[\nu] - \Lambda^{j+1}\, \hat{h}[\nu]\, \tau_{[\bar{m}],\bar{l}}[\nu]\, \omega^{\bar{\lambda}\nu}, \ \nu = -N/2, ..., N/2 - 1$.
5. Calculate the inner products

$$T_{[m],l,\lambda}(\mathbf{z}^{j+1}) = \frac{1}{N} \sum_{\nu=-N/2}^{N/2-1} \omega^{\lambda\nu}\ \hat{h}[\nu]\, \tau_{[m],l}[\nu]\, \hat{\mathbf{z}}^{j+1}[n]^*.$$

6. Calculate the oblique projection coefficients

$$\alpha_{[m],l,\lambda}(\mathbf{z}^{j+1}) = \frac{T_{[m],l,\lambda}(\mathbf{z}^{j+1})}{\left\|\theta_{[m],l}\right\|} \left(1 - \frac{e_{[m],l,\lambda}}{\left|T_{[m],l,\lambda}(\mathbf{z}^{j+1})\right|}\right)_+$$

   and the values $A_{[m],l}(\mathbf{z}^{j+1}) \stackrel{\text{def}}{=} \max |\acute{\alpha}_{[m],l,\lambda}(\mathbf{z}^{j+1})|$ with respect to $\lambda = 0, ..., N - 1$.
7. If for some $(\tilde{m}, \tilde{l})$ the values $A_{[\tilde{m}],\tilde{l}}(\mathbf{z}^{j+1}) < T_s$ then remove $\psi^{2r}_{[\tilde{m}],\tilde{l},\lambda}$ and $\theta_{[\tilde{m}],\tilde{l},\lambda}$, $\lambda = 0, ...N - 1$, from the dictionaries $\tilde{\mathbf{D}}_A$ and $\tilde{\mathbf{D}}_T$, respectively.

If the dictionaries are not empty then $j = j + 1$, go to 1. Otherwise: the solution $f[k] \approx f^j[k] = 1/\sqrt{N} \sum_{\nu=-N/2}^{N/2-1} \sigma^j[\nu]\, \omega^{k\nu}$.

Remark 14.1 The described algorithm can be utilized for restoration of signals, which are corrupted by noise (no filtering). In this case, the testing dictionary $\mathbf{D}_T$ is the same as the approximation dictionary $\mathbf{D}_A$.

The regularized Matching Pursuit is implemented by the MATLAB code `RMP_runD.m`.

Fig. 14.17 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 5$) input

14.5.3 Numerical Examples

We display a few results from application of the RMP with the discrete spline wavelet packet dictionaries to columns of the "Barbara", "Lena" and "Boats" images.

14.5.3.1 Deconvolution Examples

Restoration of a strongly blurred "Barbara" image
In this example, columns of the "Barbara" image are convolved with the Gaussian kernel whose STD $\sigma = 5$. Noise is not added.

The results of the application of the RMP algorithm to columns of the blurred image are displayed in Figs. 14.17, 14.18 and 14.19. The left frame in Fig. 14.17 displays the blurred image. Its PSNR with respect of the original image is 23.77. The image was restored by the RMT with the discrete spline wavelet packets of fourth order from first to seventh decomposition levels. Altogether, 255 waveforms were available. The threshold for the stopping rule was chosen to be 0.005. The number of waveforms in the dictionaries was reduced to 55. The right frame in Fig. 14.17 displays the restored original image. Its PSNR with respect of the original image is 26.84. We observe that the image is substantially deblurred and its fine texture is reconstructed. Figures 14.18 and 14.19 illustrate reconstruction of single columns #450 and #30, respectively. The left frames in the figures display blurred columns (solid lines) versus original columns (dashed lines). The right frames display restored columns (solid lines) versus original columns (dashed lines). One can observe that blurred columns are restored almost perfectly.

Restoration of a moderately blurred and noised "Lena" image

In this example, columns of the "Lena" image are convolved with the Gaussian kernel whose STD $\sigma = 3$ and a white noise with STD $= 5$ is added.

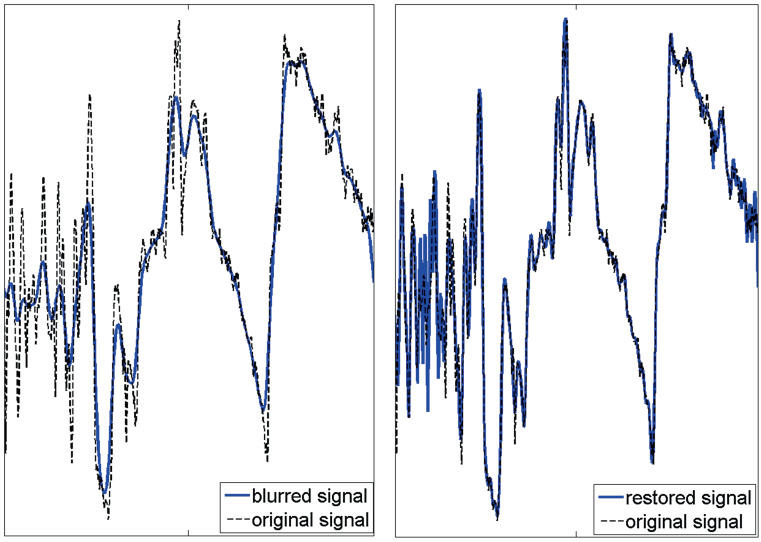


Fig. 14.18 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 5$) input. Column #450

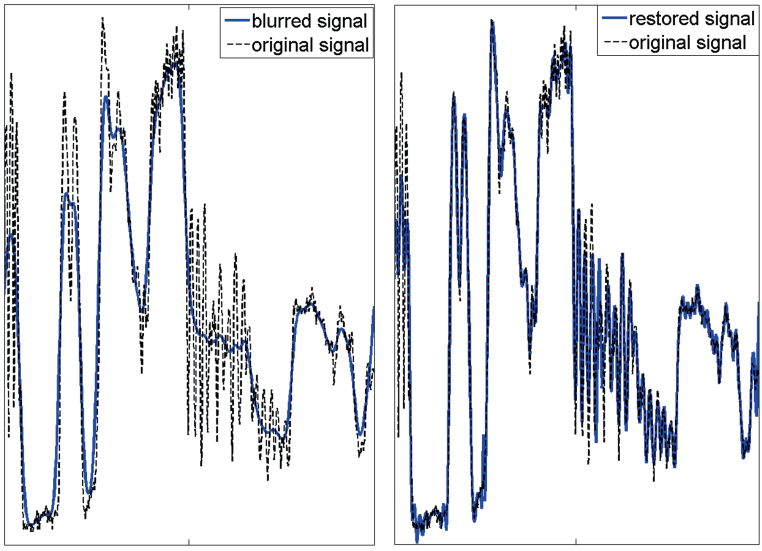


Fig. 14.19 Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 5$) input. Column #30

Fig. 14.20 Reconstruction of the "Lena" image from a blurred (Gaussian, $\sigma = 3$) and noised (STD = 5) input. *Left* input; *Right* restored image

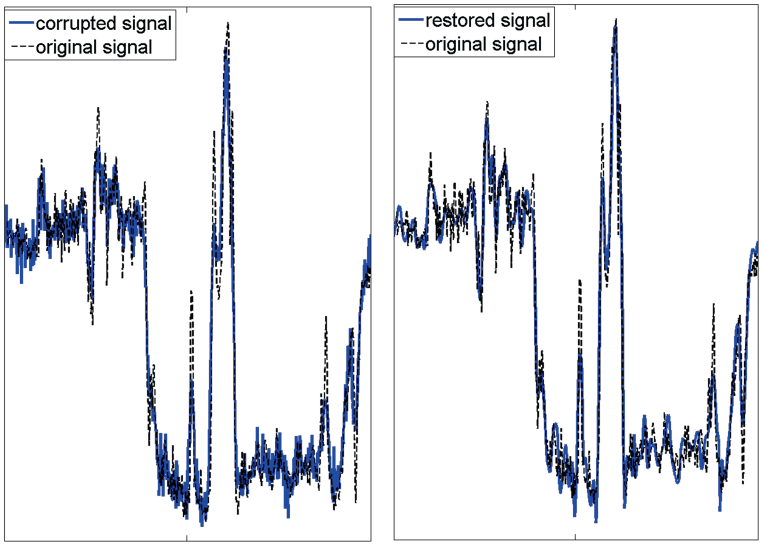


Fig. 14.21 Reconstruction of the "Lena" image from a blurred (Gaussian, $\sigma = 3$) and noised (STD = 5) input. Column #200

The results of the application of the RMP algorithm to columns of the blurred image are displayed in Figs. 14.20, 14.21 and 14.22. The PSNR of the input image is 28.26, while the PSNR of the restored image is 29.52. The image was restored by the RMT with the discrete spline wavelet packets of fourth order from first to seventh decomposition levels. Altogether, 254 waveforms were available. The threshold for the stopping rule was chosen to be 0.5. The number of waveforms in the dictionaries was reduced to 55. The image is deblurred and noise is substantially reduced. Figures 14.21 and 14.22 illustrate reconstruction of the single columns #200 and

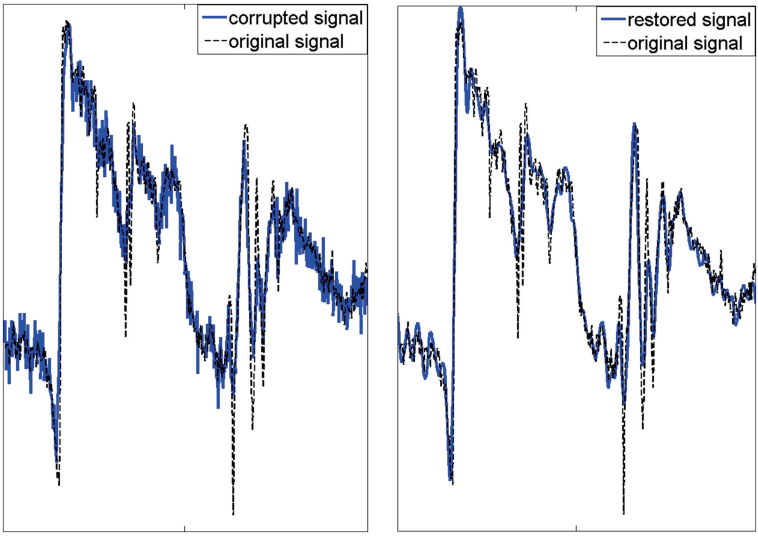


Fig. 14.22 Reconstruction of the "Lena" image from a blurred (Gaussian, $\sigma = 3$) and noised (STD = 5) input. Column #300

#300, respectively. The left frames in the figures display blurred and noised columns (solid lines) versus original columns (dashed lines). The right frames display restored columns (solid lines) versus original columns (dashed lines).

14.5.3.2 Denoising Example

The RMP algorithm proved to be efficient for restoration of signals from strongly noised input data. We illustrate its abilities on restoration of columns of the "Boats" image. In this setting, the testing dictionary is the same as the approximation one.

Restoration of a noised "Boats" image

In this example, columns of the "Boats" image are corrupted by a white noise with STD = 25. The original image is displayed in Fig. 14.23.

The results of the application of the RMP algorithm to denoising columns of the corrupted image are displayed in Figs. 14.24, 14.25 and 14.26. The PSNR of the input image is 20.18, while the PSNR of the restored image is 25.35. The image is restored by the RMT with the discrete spline wavelet packets of fourth order from first to seventh decomposition levels. The threshold for the stopping rule was chosen to be 1. Noise is substantially suppressed, while the image structure is retained. Figures 14.25 and 14.26 illustrate reconstruction of the single columns #200 and #400,respectively. The left frames in the figures display noised columns (solid lines) versus original columns (dashed lines). The right frames display restored columns

Fig. 14.23 Original "Boats" image

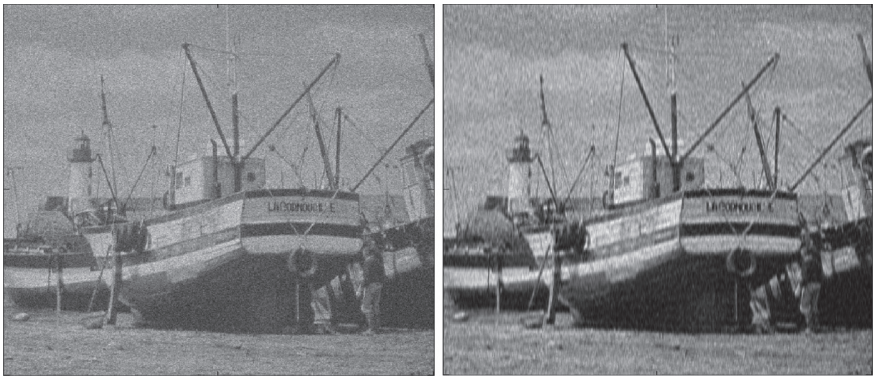


Fig. 14.24 Reconstruction of the "Boats" image from a noised (STD = 25) input

(solid lines) versus original columns (dashed lines). One can observe that the columns are restored from inputs where original signals are severely corrupted by noise.

14.5.3.3 Two Dummy Examples

In the following two examples, we assume that the *real* noise vectors **e** rather than the *estimated* vectors $\tilde{\mathbf{e}}$ are available. *Certainly, it is completely artificial setting, which never occurs in real life.* The examples are illustrations to the oblique projections method.

Dummy example: restoration of the strongly noised "Lena" image

In this example, columns of the "Lena" image are corrupted by a strong white noise with STD=200. Unlike in real life situations, for oblique projections of remainders, we use the *real* noise vectors **e** rather than the *estimated* vectors $\tilde{\mathbf{e}}$.

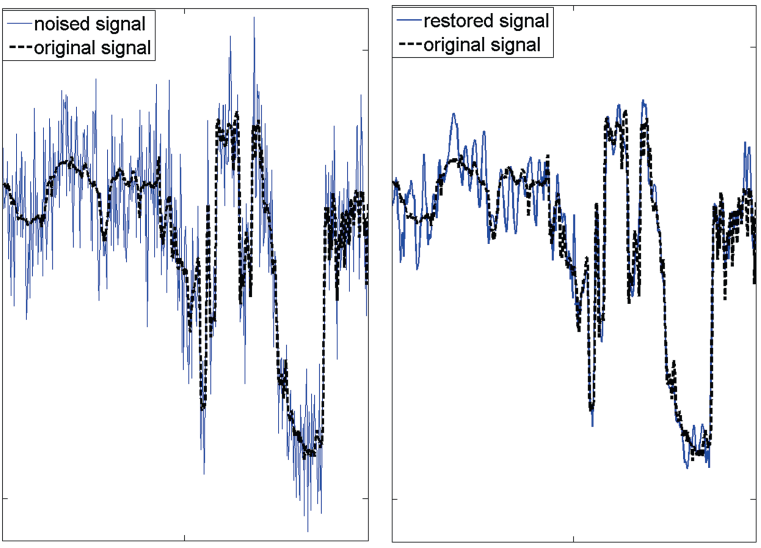


Fig. 14.25 Reconstruction of the "Boats" image from a noised (STD = 25) input. Column #200

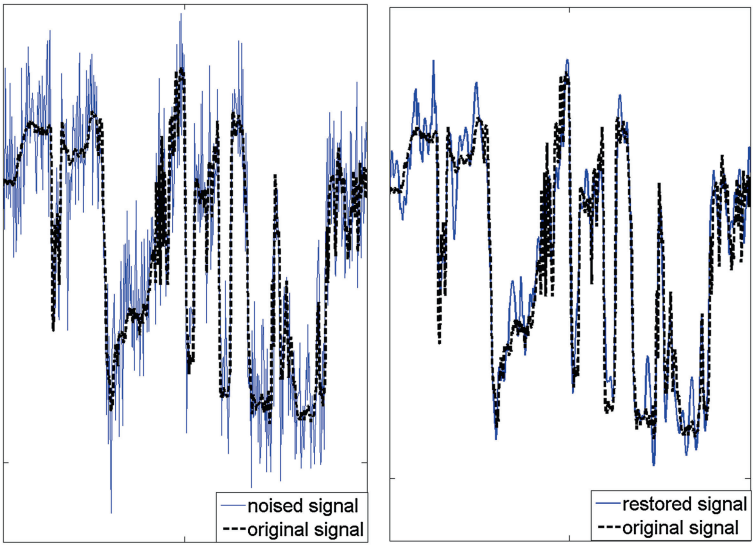


Fig. 14.26 Reconstruction of the "Boats" image from a noised (STD = 25) input. Column #400

The results of the application of the RMP algorithm to denoising the columns of the corrupted image are displayed in Figs. 14.27 and 14.28. The PSNR of the input image is 2.11, while the PSNR of the restored image is 28.41. The image is restored by the RMT with the discrete spline wavelet packets of fourth order from

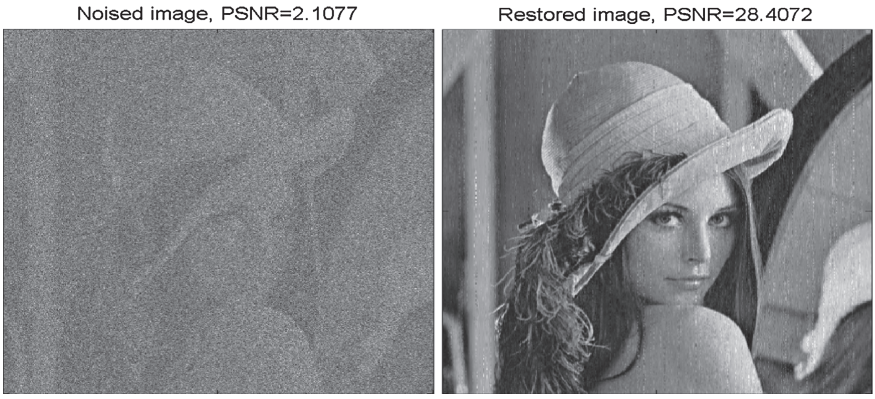


Fig. 14.27 Dummy example: reconstruction of the "Lena" image from a noised (STD = 200) input. *Left* input; *Right* restored image

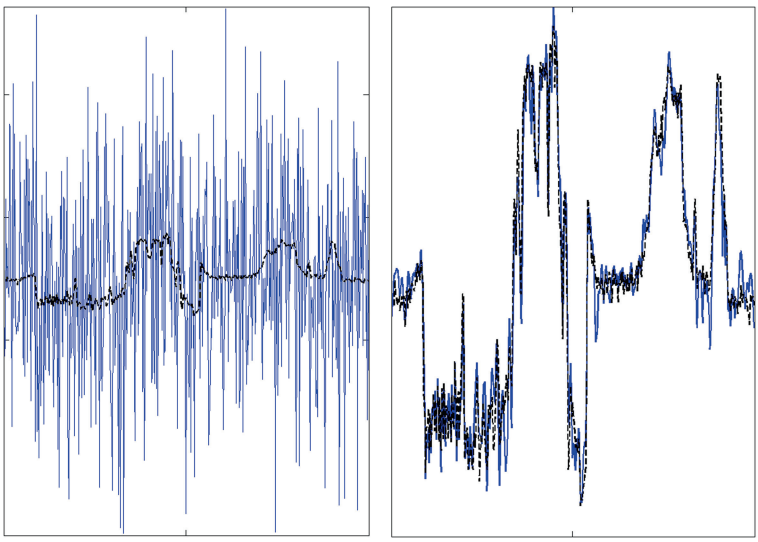


Fig. 14.28 Dummy example: reconstruction of the "Lena" image from a noised (STD = 200) input. Column #232

first to seventh decomposition levels. Altogether, 255 waveforms, whose number was reduced to 220, were available. The threshold for the stopping rule was chosen to be 10. The image is successfully restored from the input, which is completely undistinguishable. Figure 14.28 illustrates reconstruction of the single column #232. The left frame in the figure displays the noised column (solid line) versus the original column (dashed line). The right frame displays the restored column (solid line) versus the original column (dashed line). One can observe that the column is restored from

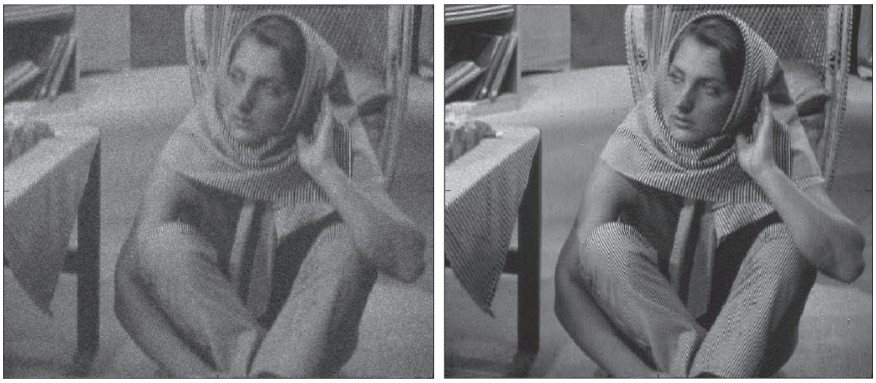


Fig. 14.29 Dummy example: Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 2$) and noised (STD = 20) input

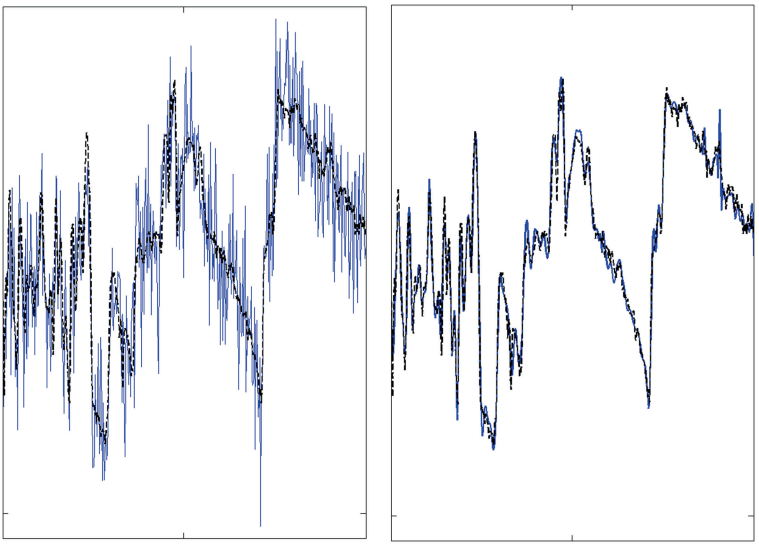


Fig. 14.30 Dummy example: Reconstruction of the "Barbara" image from a blurred (Gaussian, $\sigma = 2$) and noised (STD = 20) input. Column #450

the input where original signal is completely buried in noise. Certainly, in real life we never get to know *real* noise vectors.

Dummy example: restoration of the blurred and strongly noised "Barbara" image

In this example, the conditions of the experiment were the same as in the example demonstrated in Fig. 10.6, where the PSNR of the restored image with respect to the original one was 23.73 dB. Columns of the Barbara image (Fig. 10.5) were blurred

using Gaussian kernel with $\sigma = 2$ and corrupted by white noise whose STD was 20. For oblique projections of remainders, we use the *real* noise vectors **e** rather than the *estimated* vectors $\tilde{\mathbf{e}}$.

The results of the application of the RMP algorithm to restoration of columns of the corrupted image are displayed in Figs. 14.29 and 14.30. The PSNR of the input image is 20.87, while the PSNR of the restored image is 28.43. The image is restored by the RMT with the discrete spline wavelet packets of fourth order from first to seventh decomposition levels. Altogether, 255 waveforms were available. The threshold for the stopping rule was set to be five. The number of waveforms was reduced to 90. Noise is almost completely removed and the texture is successfully restored even in the regions where it was lost in the input. Figure 14.30 illustrates reconstruction of the single column #450. The left frame in the figure displays the noised column (solid line) versus the original column (dashed line). The right frame displays the restored column (solid line) versus the original column (dashed line). One can observe that the column is restored almost perfectly.

References

1. R.R. Coifman, V.M. Wickerhauser, Entropy-based algorithms for best basis selection. IEEE Trans. Inform. Theor. **38**(2), 713–718 (1992)
2. N. Saito, R.R. Coifman, Local discriminant bases and their applications. J. Math. Imaging Vis **5**(4), 337–358 (1995)
3. N. Saito, R.R. Coifman, in *Improved Discriminant Bases Using Empirical Probability Density Estimation*. Proceedings of the Statistical Computing Section, American Statistical Association, pp. 312–321, Washington 1997
4. M.V. Wickerhauser, in *Adapted Wavelet Analysis: From Theory to Software* (AK Peters, Wellesley, 1994)

Chapter 15
Biorthogonal Wavelet Transforms

Abstract In this chapter, design and implementation of biorthogonal wavelet transforms of periodic signals is described. For this purpose, perfect reconstruction (PR) p-filter banks are used. In particular, design of compactly supported wavelets, such as popular 5/3 and 9/7 wavelets is outlined. Adaptation of the concepts of polynomials restoration and of vanishing moments to the discrete periodic setting is discussed.

It is important for many applications of the wavelet analysis, such as data compression, denoising, deconvolution, to name a few, that the involved wavelets be (anti) symmetric. Compact support of the wavelets (FIR filter banks) is important, for example, in real time processing. The orthogonality of the wavelets is desirable because this property allows the use of the same p-filters for the analysis and the synthesis transforms. In addition, orthogonal transforms split the signal space into mutually orthogonal subspaces.

However, it is well known that the orthogonallity of wavelets is incompatible with the requirements of symmetry and compact support [1]. The only exception is the Haar wavelets. All other orthogonal symmetric wavelet systems consist of non-compactly supported wavelets. The periodic spline orthogonal wavelet and wavelet packet transforms are described in Chaps. 8, 9 and 14.

In order to keep the compact support and the symmetry requirement, the orthogonality should be relaxed to the biorthogonality. A general approach to the design of compactly supported biorthogonal wavelets, which utilized the Bezout theorem, was developed in [2] and [3]. In addition, a variety of examples of such wavelets, many of whom are linked to polynomial splines, are presented in [2] and [3]. One of these wavelets, the so-called 9/7 wavelet, which is presented in [2] (see also [4]) proved to be especially efficient in image compression. It is included, together with the so-called 5/3 wavelet into the JPEG 2000 image compression standard [5].

In turn, dismissal of the compact support limitation provides additional flexibility to the design of wavelet transforms. In Chap. 16, we describe how polynomial and discrete splines can serve as sources for design of a family of biorthogonal wavelet bases in the signal space. Although the wavelets to be designed originate

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_15,

from splines, they are not splines themselves. Properties of the designed wavelets, such as symmetry, flat spectra, good time domain localization, vanishing moments, and smoothness of the waveforms, are valuable for signal processing.

15.1 Two-Channel Filter Banks

It was indicated in Sect. 2.2.1 that critically sampled perfect reconstruction (PR) p-filter banks produce biorthogonal bases in the signal space $\Pi[N]$. In this section we outline two channel p-filter banks with downsampling factor of 2.

15.1.1 Matrix Expression of p-Filter Banks

Assume that $\tilde{\mathbf{H}} = \left\{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1\right\}$ is an analysis p-filter bank with downsampling factor of 2, where $\tilde{\mathbf{h}}^0 = \left\{\tilde{h}^0[k]\right\} \in \Pi[N]$ is a low-pass filter, while $\tilde{\mathbf{h}}^1 = \left\{\tilde{h}^1[k]\right\} \in \Pi[N]$ is a high-pass p-filter. Application of the time-reversed p-filter bank $\tilde{\mathbf{H}}$ to a signal $\mathbf{x} = \{x[k]\} \in \Pi[N]$ produces two signals, $\mathbf{y}^0$ and $\mathbf{y}^1$, from space $\Pi[N/2]$:

$$y^0[l] = \textstyle\sum_{k=0}^{N-1} \tilde{h}^0[k-2l]\, x[k],$$

$$y^1[l] = \textstyle\sum_{k=0}^{N-1} \tilde{h}^1[k-2l]\, x[k], \quad l \in \mathbb{Z}.$$

Denote $\check{\mathbf{h}}^0 = \left\{\tilde{h}^0[-k]\right\} \in \Pi[N]$. The DFT of the signal $\check{\mathbf{h}}^0$, which is the frequency response (FR) of the p-filter,

$$\hat{\check{h}}^0[n] = \sum_{k=0}^{N-1} \omega^{-kn}\, \tilde{h}^0[-k] = \hat{\tilde{h}}^0[n]^* = \hat{\tilde{h}}^0[-n].$$

The signal $\mathbf{y}^0$ can be regarded as the zero polyphase component of the discrete convolution $y^0[l] = \sum_{k=0}^{N-1} \check{h}^0[\lambda - k]\, x[k]\,|_{\lambda=2l}$. Thus, the DFT

$$\hat{y}^0[n]_1 \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} \omega^{-2ln}\, y^0[l] = \frac{1}{2}\left(\hat{\tilde{h}}^0[-n]\,\hat{x}[n] + \hat{\tilde{h}}^0[-n+N/2]\,\hat{x}[n+N/2]\right).$$

Similarly,

$$\hat{y}^1[n]_1 = \frac{1}{2}\left(\hat{\tilde{h}}^1[-n]\,\hat{x}[n] + \hat{\tilde{h}}^1[-n+N/2]\,\hat{x}[n+N/2]\right).$$

The latter relations can be expressed in a matrix form

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix} = \frac{1}{2}\tilde{\mathbf{M}}[-n] \cdot \begin{pmatrix} \hat{x}[n] \\ \hat{x}[n+N/2] \end{pmatrix}, \text{ where } \tilde{\mathbf{M}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}^0[n] & \hat{\tilde{h}}^0[n+N/2] \\ \hat{\tilde{h}}^1[n] & \hat{\tilde{h}}^1[n+N/2] \end{pmatrix}$$

is the analysis modulation matrix.

Let $\mathbf{x}_m \stackrel{\text{def}}{=} \{x[2k+m]\}$, $k \in \mathbb{Z}$, $m = 0, 1$ be polyphase components of the signal $\mathbf{x}$ and $\mathbf{h}^p_m \stackrel{\text{def}}{=} \{h^p[2k+m]\}$, $k \in \mathbb{Z}$, $m = 0, 1$ be polyphase components of the p-filters $\mathbf{h}^p$, $p = 0, 1$. The DFT of output signals can be presented in a polyphase form

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix} = \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}, \text{ where } \tilde{\mathbf{P}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}^0_0[n]_1 & \hat{\tilde{h}}^0_1[n]_1 \\ \hat{\tilde{h}}^1_0[n]_1 & \hat{\tilde{h}}^1_1[n]_1 \end{pmatrix} \quad (15.1)$$

is the analysis polyphase matrix.

Assume that $\mathbf{H} = \{\mathbf{h}^0, \mathbf{h}^1\}$ is a synthesis p-filter bank with upsampling factor of 2, where $\mathbf{h}^0 = \{h^0[k]\} \in \Pi[N]$ is a low-pass filter, while $\mathbf{h}^1 = \{h^1[k]\} \in \Pi[N]$ is a high-pass filter. Application of the p-filter bank $\mathbf{H}$ to signals $\mathbf{y}^0$ and $\mathbf{y}^1$ produces a signal $\check{\mathbf{x}}$

$$\check{x}[k] = \sum_{l=0}^{N-1} h^0[k-2l]\, y^0[l] + h^1[k-2l]\, y^1[l] \quad (15.2)$$

$$\hat{\check{x}}[n] = \hat{y}^0[2n]\, \hat{h}^0[n] + \hat{y}^1[2n]\, \hat{h}^1[n]. \quad (15.3)$$

The relations Eq. (15.3) can be expressed in a matrix form:

$$\begin{pmatrix} \hat{\check{x}}[n] \\ \hat{\check{x}}[n+N/2] \end{pmatrix} = \mathbf{M}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix}, \text{ where } \mathbf{M}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}^0[n] & \hat{h}^1[n] \\ \hat{h}^0[n+N/2] & \hat{h}^1[n+N/2] \end{pmatrix}$$

is the synthesis modulation matrix.

The DFT of the signal $\check{\mathbf{x}}$ can be presented in a polyphase form

$$\begin{pmatrix} \hat{\check{x}}_0[n]_1 \\ \hat{\check{x}}_1[n]_1 \end{pmatrix} = \mathbf{P}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix}, \text{ where } \mathbf{P}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}^0_0[n]_1 & \hat{h}^1_0[n]_1 \\ \hat{h}^0_1[n]_1 & \hat{h}^1_1[n]_1 \end{pmatrix} \quad (15.4)$$

is the synthesis polyphase matrix.

Definition 15.1 Assume that the pair of signals $\mathbf{y}^0$ and $\mathbf{y}^1$ are derived by application of the analysis p-filter bank $\tilde{\mathbf{H}}$ to a signal $\mathbf{x}$, and application of the synthesis p-filter bank $\mathbf{H}$ to the signals $\mathbf{y}^0$ and $\mathbf{y}^1$ restores the signal $\mathbf{x}$. If this is true for any signal $\mathbf{x} \in \Pi[N]$ then the pair $\tilde{\mathbf{H}}$ and $\mathbf{H}$ constitutes a perfect reconstruction (PR) p-filter bank.

A PR p-filter bank is characterized by the matrix relations

$$\mathbf{M}[n] \cdot \tilde{\mathbf{M}}[-n] = 2\mathbf{I}_2, \quad \mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \mathbf{I}_2, \quad \mathbf{I}_2 \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{15.5}$$

Equation (15.5) imply, in particular, the following relations between the analysis and synthesis p-filters:

$$\begin{aligned} &\hat{h}0[n]\,\hat{\tilde{h}}^0[-n] + \hat{h}^1[n]\,\hat{\tilde{h}}^1[-n] = 2, \\ &\hat{h}^0[n]\,\hat{\tilde{h}}^0[-n+N/2] + \hat{h}^1[n]\,\hat{\tilde{h}}^1[-n+N/2] = 0. \end{aligned} \tag{15.6}$$

15.1.2 Biorthogonal Bases Generated by PR p-Filter Banks

It follows from Eq. (15.5) that the matrix $\mathbf{M}[n] = \left(\tilde{\mathbf{M}}[-n]\right)^{-1}$, which implies the "perfect decomposition" property:

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix} = \frac{1}{2}\,\tilde{\mathbf{M}}[-n] \cdot \mathbf{M}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \hat{y}^1[n]_1 \end{pmatrix} \tag{15.7}$$

Hence, the following relations between p-filters arise

$$\hat{h}^i[n]\,\hat{\tilde{h}}^j[-n] + \hat{h}^i[n+N/2]\,\hat{\tilde{h}}^j[-n+N/2] = 2\,\delta[i-j],\ i,j = 0,1. \tag{15.8}$$

Proposition 15.1 *Assume that the pair* $\tilde{\mathbf{H}}$ *and* $\mathbf{H}$ *constitute a PR p-filter bank: thus the conditions* Eqs. (15.5) *and* (15.7) *are satisfied. Then two-sample shifts* $\left\{\tilde{h}^0[\cdot - 2l],\ \tilde{h}^1[\cdot - 2l]\right\}$ *and* $\left\{h^0[\cdot - 2l],\ h^1[\cdot - 2l]\right\}$, $k,l = 0,\ldots,N/2-1$, *of impulse responses (IR) of the analysis and synthesis p-filter banks* $\tilde{\mathbf{H}}$ *and* $\mathbf{H}$*, respectively, form a biorthogonal basis of the signal space* $\Pi[N]$.

Proof Since $\tilde{\mathbf{H}}$ and $\mathbf{H}$ constitute a PR p-filter bank, any signal $\mathbf{x} \in \Pi[N]$ can be represented as in Eq. (15.2)

$$x[k] = \sum_{l\in\mathbb{Z}} h^0[k-2l]\,y^0[l] + h^1[k-2l]\,y^1[l], \text{ where}$$

$$y^p[l] = \sum_{k=0}^{N-1} \tilde{h}^p[k-2l]\,x[k] = \left\langle \mathbf{x}, \tilde{\mathbf{h}}^p[\cdot - 2l] \right\rangle, \quad p = 0,1. \tag{15.9}$$

On the other hand, Eq. (15.7) and its derivative Eq. (15.8) imply that the inner products of the presumed basis signals are

$$\begin{aligned}\left\langle \mathbf{h}^i[\cdot - 2\lambda], \tilde{\mathbf{h}}^i[\cdot - 2l] \right\rangle &= \sum_{k=0}^{N-1} h^i[k - 2\lambda]\, \tilde{h}^i[k - 2l]^* \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \omega^{2n(\lambda - l)}\, \hat{h}^i[n]\, \hat{\tilde{h}}^i[-n] \\ &\quad \times \frac{1}{N} \sum_{n=0}^{N/2-1} \omega^{2n(\lambda - l)} \left(\hat{h}^i[n]\, \hat{\tilde{h}}^i[-n] + \hat{h}^i[n + N/2]\, \hat{\tilde{h}}^i[-n + N/2] \right) \\ &= \frac{2}{N} \sum_{n=0}^{N/2-1} \omega^{2n(\lambda - l)} = \delta[\lambda - l].\end{aligned}$$

Here $i = 0, 1$. In addition, Eq. (15.8) implies that

$$\left\langle \mathbf{h}^0[\cdot - 2\lambda], \tilde{\mathbf{h}}^1[\cdot - 2l] \right\rangle = \left\langle \mathbf{h}^1[\cdot - 2\lambda], \tilde{\mathbf{h}}^0[\cdot - 2l] \right\rangle = 0.$$

Thus, the proposition's claim follows. ■

Remark 15.1.1 According to Proposition 15.1, the PR p-filter bank $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$ can be referred to as the biorthogonal p-filter bank. The analysis $\left\{\tilde{\mathbf{h}}^0[\cdot - 2l],\ \tilde{\mathbf{h}}^1[\cdot - 2l]\right\}$ and the synthesis $\left\{\mathbf{h}^0[\cdot - 2l],\ \mathbf{h}^1[\cdot - 2l]\right\}$, $l = 0, \ldots, N/2$, biorthogonal systems are interchangeable. A signal $\mathbf{x} \in \Pi[N]$ can be represented as

$$\begin{aligned}x[k] &= \sum_{l \in \mathbb{Z}} \tilde{h}^0[k - 2l]\, \tilde{y}^0[l] + \tilde{h}^1[k - 2l]\, \tilde{y}^1[l], \text{ where} \\ \tilde{y}^i[l] &= \sum_{k=0}^{N-1} h^i[k - 2l]\, x[k] = \left\langle \mathbf{x}, \mathbf{h}^i[\cdot - 2l] \right\rangle, \quad p = 0, 1.\end{aligned}$$

Proposition 15.2 *If there exists abiorthogonal pair* $\left\{\tilde{\mathbf{h}}^0,\ \mathbf{h}^0\right\}$ *of the low-pass p-filters, such that* Eq. (15.8) *holds with* $i = j = 0$*, then the complementary high-pass p-filters can be defined via the DFT:*

$$\hat{h}^1[n] \stackrel{\text{def}}{=} \omega^n\, \hat{\tilde{h}}^0[-n + N/2], \quad \hat{\tilde{h}}^1[n] \stackrel{\text{def}}{=} \omega^n\, \hat{h}^0[-n + N/2]. \tag{15.10}$$

Then, the pair $\mathbf{H} = \left\{\mathbf{h}^0,\ \mathbf{h}^1\right\}$ *and* $\tilde{\mathbf{H}} = \left\{\tilde{\mathbf{h}}^0,\ \tilde{\mathbf{h}}^1\right\}$ *constitutes a biorthogonal p-filter bank.*

Proof If Eq. (15.8) holds with $i = j = 0$ and the DFT of p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ are defined by Eq. (15.10) then the PR relations

$$\begin{aligned}
&\hat{h}^0[n]\,\hat{\tilde{h}}^0[-n] + \hat{h}^1[n]\,\hat{\tilde{h}}^1[-n] \\
&\quad = \hat{h}^0[n]\,\hat{\tilde{h}}^0[-n] + \hat{h}^0[n+N/2]\,\hat{\tilde{h}}^0[-n+N/2] = 2, \\
&\hat{h}^0[n]\,\hat{\tilde{h}}^0[-n+N/2] + \hat{h}^1[n]\,\hat{\tilde{h}}^1[-n+N/2] \\
&\quad = \hat{h}^0[n]\,\hat{\tilde{h}}^0[-n+N/2] - \hat{h}^0[n]\,\hat{\tilde{h}}^0[-n+N/2] = 0
\end{aligned}$$

take place. Due to Proposition 15.1, the same is true for the biorthogonal relations:

$$\begin{aligned}
\hat{h}^1[n]\,\hat{\tilde{h}}^1[-n] + \hat{h}^1[n+N/2]\,\hat{\tilde{h}}^1[-n+N/2] = 2, \\
\hat{h}^1[n]\,\hat{\tilde{h}}^0[-n] + \hat{h}^1[n+N/2]\,\hat{\tilde{h}}^0[-n+N/2] = 0.
\end{aligned}$$

■

Corollary 15.1 *If the conditions of Proposition* 15.2 *are satisfied then the impulse responses of the high-pass p-filters are linked with the impulse responses of the low-pass p-filters as*

$$h^1[k] = (-1)^k \tilde{h}^0[-k-1], \quad \tilde{h}^1[k] = (-1)^k h^0[-k-1], \ k \in \mathbb{Z}. \tag{15.11}$$

15.1.3 Multilevel Wavelet Transforms

Assume that a pair $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$ forms a biorthogonal p-filter bank. Then the impulse responses of the synthesis and analysis p-filters generate a biorthogonal basis of the signal space $\Pi[N]$. Denote,

$$\tilde{\psi}^s_{[1]}[k] \stackrel{\text{def}}{=} \tilde{h}^s[k], \quad \psi^s_{[1]}[k] \stackrel{\text{def}}{=} h^s[k], \ s = 0, 1, \ k \in \mathbb{Z},$$

and $\mathbf{y}^0_{[1]} \stackrel{\text{def}}{=} \mathbf{y}^0, \mathbf{y}^1_{[1]} \stackrel{\text{def}}{=} \mathbf{y}^1$.

The periodic signals $\tilde{\psi}^0_{[1]}[k]$ and $\tilde{\psi}^1_{[1]}[k]$ are referred to as the discrete-time low-frequency and high-frequency analysis wavelets, respectively, of the first decomposition level while $\psi^0_{[1]}[k]$ and $\psi^1_{[1]}[k]$ are called synthesis wavelets. Then, expansion of a signal $\mathbf{x} \in \Pi[N]$ becomes $\mathbf{x} = \mathbf{x}^0_{[1]} + \mathbf{x}^1_{[1]}$, where

$$x^s_{[1]}[k] = \sum_{l=0}^{N/2-1} y^s_{[1]}[l]\,\psi^s_{[1]}[k-2l], \quad s = 0, 1, \tag{15.12}$$

$$y^s_{[1]}[l] = \sum_{k=0}^{N-1} \tilde{h}^s_{[1]}[k-2l]\, x[k] \tag{15.13}$$
$$= \sum_{k=0}^{N-1} \tilde{\psi}^s_{[1]}[k-2l]\, x[k] = \left\langle \mathbf{x},\, \tilde{\psi}^s_{[1]}[\cdot - 2l] \right\rangle.$$

Denote, by $\tilde{\mathbf{H}}_{[1]} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s_{[1]}\right\}$, $s = 0, 1$, an analysis, and by $\mathbf{H}_{[1]} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s_{[1]}\right\}$, $s = 0, 1$, a a synthesis p-filter bank, which operate in the subspace $\Pi[N/2]$. Assume that the pair $\left\{\tilde{\mathbf{H}}_{[1]},\, \mathbf{H}_{[1]}\right\}$ form a PR p-filter bank in the space $\Pi[N/2]$. It means,in particular, that the IR of the p-filters $\left\{\tilde{h}^s_{[1]}[k]\right\}$ and $\left\{h^s_{[1]}[k]\right\}$, $k \in \mathbb{Z}$, $s = 0, 1$, are $N/2$-periodic signals.

Remark 15.1.2 It is natural to define the p-filter banks $\tilde{\mathbf{H}}_{[1]}[n]$ and $\mathbf{H}_{[1]}[n]$ via their polyphase matrices as $\tilde{\mathbf{P}}_{[1]}[n] \stackrel{\text{def}}{=} \tilde{\mathbf{P}}[2n]$, $\mathbf{P}_{[1]}[n] \stackrel{\text{def}}{=} \mathbf{P}[2n]$, where $\tilde{\mathbf{P}}[n]$ and $\mathbf{P}[n]$, $n \in \mathbb{Z}$, are the polyphase matrices of the p-filter banks $\tilde{\mathbf{H}}$ and $\mathbf{H}$, respectively. Surely, the PR conditions $\mathbf{P}_{[1]}[n] \cdot \tilde{\mathbf{P}}_{[1]}[-n] = \mathbf{I}_2$ are satisfied. However, different p-filter banks are possible.

Similarly to the whole signal $\mathbf{x}$, the low-frequency signal $\mathbf{y}^0_{[1]} \in \Pi[N/2]$ can be represented as

$$y^0_{[1]}[k] = \sum_{l=0}^{N/4-1} y^0_{[2]}[l]\, h^0_{[1]}[k-2l] + y^1_{[2]}[l]\, h^1_{[1]}[k-2l], \tag{15.14}$$

where the coefficients are

$$y^s_{[2]}[l] = \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu - 2l]\, y^0_{[1]}[\nu] = \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu - 2l] \sum_{\kappa=0}^{N-1} \tilde{\psi}^0_{[1]}[\kappa - 2\nu]\, x[\kappa]$$
$$= \sum_{\kappa=0}^{N-1} x[\kappa] \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu - 2l]\, \tilde{\psi}^0_{[1]}[\kappa - 2\nu]$$
$$= \sum_{\kappa=0}^{N-1} x[\kappa] \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu]\, \tilde{\psi}^0_{[1]}[\kappa - 2\nu - 4l], \quad s = 0, 1.$$

Denote,

$$\tilde{\psi}^{s}_{[2]}[\kappa] \stackrel{\text{def}}{=} \sum_{\nu=0}^{N/2-1} \tilde{h}^{s}_{[1]}[\nu]\, \tilde{\psi}^{0}_{[1]}[\kappa - 2\nu].$$

Then, the coefficients are represented via the inner products: $y^{s}_{[2]}[l] = \left\langle \mathbf{x},\, \tilde{\psi^{s}_{[2]}}[\cdot - 4l] \right\rangle$.

By substituting Eq. (15.14) into Eq. (15.12) with $s = 0$, we get

$$\begin{aligned}
x^{0}_{[1]}[k] &= \sum_{l=0}^{N/2-1} y^{0}_{[1]}[l]\, \psi^{0}_{[1]}[k - 2l] \\
&= \sum_{l=0}^{N/2-1} \psi^{0}_{[1]}[k - 2l] \sum_{s=0}^{1} \sum_{\lambda=0}^{N/4-1} y^{s}_{[2]}[\lambda]\, h^{s}_{[1]}[l - 2\lambda] \\
&= \sum_{s=0}^{1} \sum_{\lambda=0}^{N/4-1} y^{s}_{[2]}[\lambda] \sum_{l=0}^{N/2-1} \psi^{0}_{[1]}[k - 2l]\, h^{s}_{[1]}[l - 2\lambda] \\
&= \sum_{s=0}^{1} \sum_{\lambda=0}^{N/4-1} y^{s}_{[2]}[\lambda]\; \psi^{s}_{[2]}[k - 4\lambda],
\end{aligned}$$

where

$$\psi^{s}_{[2]}[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} \psi^{0}_{[1]}[k - 2l]\, h^{s}_{[1]}[l],\ s = 0, 1.$$

The signals $\left\{\tilde{\psi}^{s}_{[2]}\right\}$ and $\left\{\psi^{s}_{[2]}\right\}$, $s = 0, 1$, are called the analysis and the synthesis discrete-time periodic wavelets of the second decomposition level, respectively.

As a result from the two-level wavelet transform, we have the signal $\mathbf{x}$ expanded over 4-sample shifts of the second level wavelets and 2-sample shifts of the first level high-frequency wavelets: $\mathbf{x} = \sum_{s=0}^{1} \mathbf{x}^{s}_{[2]} + \mathbf{x}^{1}_{[1]}$,
where

$$x^{s}_{[2]}[k] = \sum_{l=0}^{N/4-1} y^{s}_{[2]}[l]\, \psi^{s}_{[2]}[k - 4l], \quad x^{1}_{[1]}[k] = \sum_{l=0}^{N/2-1} y^{1}_{[1]}[l]\, \psi^{1}_{[1]}[k - 2l],$$

$$y^{s}_{[2]}[l] = \left\langle \mathbf{x},\, \tilde{\psi}^{s}_{[2]}[\cdot - 4l] \right\rangle, \quad y^{1}_{[1]}[l] = \left\langle \mathbf{x},\, \tilde{\psi}^{1}_{[1]}[\cdot - 2l] \right\rangle,\ s = 0, 1.$$

Assume that the analysis $\tilde{\mathbf{H}}_{[\mu]} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^{s}_{[\mu]}\right\}$, $s = 0, 1$, and the synthesis $\mathbf{H}_{[\mu]} \stackrel{\text{def}}{=} \left\{\mathbf{h}^{s}_{[\mu]}\right\}$, $s = 0, 1$, p-filter banks with downsampling by factor of 2, which operate in the subspaces $\Pi[N/2^{\mu}]$, form PR p-filter banks and $\tilde{\mathbf{P}}_{[\mu]}[n]$, $\mu = 0, \ldots, -1$, and $\mathbf{P}_{[\mu]}[n]$ are their polyphase matrices, respectively.

Then, iterated transforms with the p-filter banks $\tilde{\mathbf{H}}_{[\mu]}[n]$ and $\mathbf{H}_{[\mu]}[n]$

$$y^0_{[\mu]}[k] = \sum_{l=0}^{N/2^{\mu+1}-1} y^0_{[\mu+1]}[l]\, h^0_{[\mu]}[k-2l] + y^1_{[\mu+1]}[l]\, h^1_{[\mu]}[k-2l], \tag{15.15}$$

$$y^s_{[\mu+1]}[l] = \sum_{\nu=0}^{N/2^{\mu}-1} \tilde{h}^s_{[\mu]}[\nu-2l]\, y^0_{[\mu]}[\nu], \quad s=0,1,\ \mu=1,\dots,m-1, \tag{15.16}$$

leads to wavelet expansion of the signal $\mathbf{x} = \mathbf{x}^0_{[m]} + \sum_{\mu=1}^{m} \mathbf{x}^1_{[\mu]}$, where

$$\begin{aligned} x^s_{[m]}[k] &= \textstyle\sum_{l=0}^{N/2^m-1} y^s_{[m]}[l]\, \psi^s_{[m]}[k-2^m l],\ s=0,1,\\ x^1_{[\mu]}[k] &= \textstyle\sum_{l=0}^{N/2^\mu-1} y^1_{[\mu]}[l]\, \psi^1_{[\mu]}[k-2^\mu l], \quad \mu<m,\\ y^s_{[m]}[l] &= \left\langle \mathbf{x}, \tilde{\psi}^s_{[m]}[\cdot-2^m l]\right\rangle, \quad y^1_{[\mu]}[l] = \left\langle \mathbf{x}, \tilde{\psi}^1_{[\mu]}[\cdot-2^\mu l]\right\rangle. \end{aligned} \tag{15.17}$$

The synthesis and analysis wavelets are derived iteratively

$$\begin{aligned} \psi^s_{[\mu]}[k] &\stackrel{\text{def}}{=} \textstyle\sum_{l=0}^{N/2^{\mu-1}-1} h^s_{[\mu-1]}[l]\, \psi^0_{[\mu-1]}[k-2^{\mu-1}l],\\ \psi^{\tilde{s}}_{[\mu]}[k] &\stackrel{\text{def}}{=} \textstyle\sum_{l=0}^{N/2^{\mu-1}-1} \tilde{h}^s_{[\mu-1]}[l]\, \tilde{\psi}^0_{[\mu-1]}[k-2^{\mu-1}l],\ s=0,1. \end{aligned} \tag{15.18}$$

Proposition 15.3 *Assume that $N = 2^j$ and the integer $m < j$. The wavelet systems*

$$\tilde{\Psi}_m \stackrel{\text{def}}{=} \left\{\tilde{\psi}^0_{[m]}[k-2^m l]\right\}_{l=0}^{N/2^m} \bigcup \left\{ \biguplus_{\mu=1}^{m} \tilde{\psi}^1_{[\mu]}[k-2^\mu l] \right\}_{l=0}^{N/2^\mu},$$

$$\Psi_m \stackrel{\text{def}}{=} \left\{\psi^0_{[m]}[k-2^m l]\right\}_{l=0}^{N/2^m} \bigcup \left\{ \biguplus_{\mu=1}^{m} \psi^1_{[\mu]}[k-2^\mu l] \right\}_{l=0}^{N/2^\mu}$$

form a biorthogonal pair of bases in the signal space $\Pi[N]$.

Proof Note that either of the systems $\tilde{\Psi}_m$ and Ψ_m comprises exactly N terms. Denote elements of the system $\tilde{\Psi}_m$ as $\tilde{\mathbf{b}}_\nu$ and the respective elements of Ψ_m as $\mathbf{b}_\nu$, $\nu = 1,\dots,N$. Equation (15.17) implies that any signal $\mathbf{x} \in \Pi[N]$ is represented as

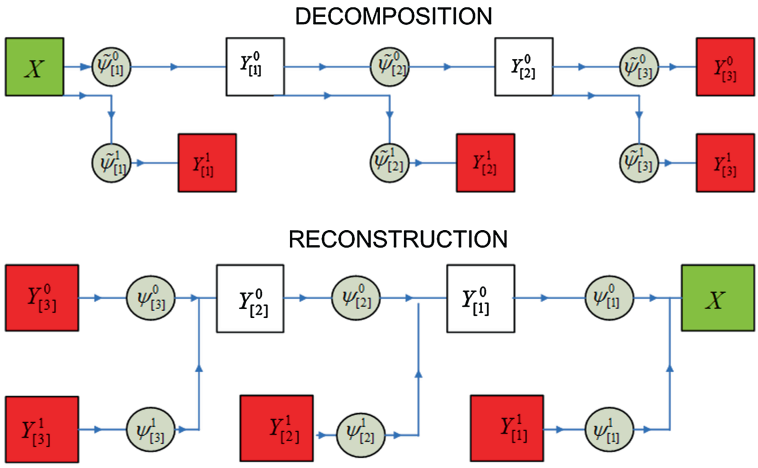


Fig. 15.1 Diagrams of direct (*top*) and inverse (*bottom*) 3-level wavelet transforms

$\mathbf{x} = \sum_{\nu=1}^{N} \left\langle \mathbf{x}, \tilde{\mathbf{b}}_\nu \right\rangle \mathbf{b}_\nu$. Thus, the system Ψ_m constitutes a basis in $\Pi[N]$. Take $\mathbf{x} = \mathbf{b}_\kappa$. Then,

$$\mathbf{b}_\kappa = \sum_{\nu=1}^{N} \left\langle \mathbf{b}_\kappa, \tilde{\mathbf{b}}_\nu \right\rangle \mathbf{b}_\nu \Longrightarrow \left\langle \mathbf{b}_\kappa, \tilde{\mathbf{b}}_\nu \right\rangle = \delta[\kappa - \nu].$$

Surely, the systems $\tilde{\Psi}_m$ and Ψ_m are interchangeable. ■

The m-level wavelet transform of a signal $\mathbf{x} \in \Pi[N]$ results in the set of N coefficients

$$\mathbf{x} \longrightarrow \mathbf{Y}_m = \mathbf{Y}^0_{[m]} \bigoplus_{\mu=1}^{m} \mathbf{Y}^1_{[\mu]}, \quad \mathbf{Y}^0_{[m]} \stackrel{\text{def}}{=} \left\{ y^0_{[m]}[l] \right\}_{l=0}^{N/2^m}, \quad \mathbf{Y}^1_{[\mu]} \stackrel{\text{def}}{=} \left\{ y^1_{[\mu]}[l] \right\}_{l=0}^{N/2^m},$$

where the coefficients $y^s_{[\mu]}[l]$ are derived by the convolution operations of Eqs. (15.12), (15.13), (15.15). The set $\mathbf{Y}_m$ of the transform coefficients can be regarded as the set of coordinates of the signal $\mathbf{x}$ in a biorthogonal basis.

The transform is invertible. Once the set $\mathbf{Y}_m$ of coefficients is available, consecutive application of the convolution operations Eqs. (15.12), (15.15) restores the signal $\mathbf{x}$. Diagrams in Fig. 15.1 illustrate direct and inverse multilevel wavelet transforms.

15.1.4 Implementation of Multi-Level Wavelet Transforms

In order to practically implement a wavelet transform of a periodic signal, it is advisable to use the DFT and the polyphase representations of the transforms,

which extends Eqs. (15.1) and (15.4). For uniformity, denote temporarily the PR pair of p-filter banks $\tilde{\mathbf{H}}$ and $\mathbf{H}$ from the initial level as $\tilde{\mathbf{H}}_{[0]}$ and $\mathbf{H}_{[0]}$, respectively. Their polyphase matrices are denoted as $\tilde{\mathbf{P}}_{[0]}[n]$ and $\mathbf{P}_{[0]}[n]$. Assume that $\mathbf{x} = \{x[k]\} \in \Pi[N]$ is a signal to be operated by the m-level wavelet transform with the PR pairs $\tilde{\mathbf{H}}_{[\mu]}$ and $\mathbf{H}_{[\mu]}$, $\mu = 0, \ldots, m-1$. For simplicity, assume that the polyphase matrices of the p-filter banks from the adjacent levels are linked as $\tilde{\mathbf{P}}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \tilde{\mathbf{P}}_{[\mu]}[2n]$, $\mathbf{P}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \mathbf{P}_{[\mu]}[2n]$.

Algorithm 1 Direct transform

1: Initialization: $\mathbf{y} \stackrel{\text{def}}{=} \mathbf{x}$. Split $\mathbf{y}$ into the even and the odd polyphase components $\mathbf{y} = \mathbf{y}_0 + \mathbf{y}_1$ and calculate the DFT of $\{\hat{y}_0[n]_1\}$ and $\{\hat{y}_1[n]_1\}$.

2: For $\mu = 1$ to m Implement the one-level transform

$$\begin{pmatrix} \hat{y}^0_{[\mu]}[n]_\mu \\ \hat{y}^1_{[\mu]}[n]_\mu \end{pmatrix} = \tilde{\mathbf{P}}_{[\mu-1]}[-n] \cdot \begin{pmatrix} \hat{y}_0[n]_\mu \\ \hat{y}_1[n]_\mu \end{pmatrix}$$

Calculate the IDFT

$$y^1_{[\mu]}[k] = \frac{2^\mu}{N} \sum_{n=0}^{N/2^\mu - 1} \omega^{2^\mu kn}\, \hat{y}^1_{[\mu]}[n]_\mu$$

and store the transform coefficients $\mathbf{y}^1_{[\mu]} \stackrel{\text{def}}{=} \left\{ y^1_{[\mu]}[k] \right\}$, $k = 0, \ldots, N/2^\mu - 1$, of the μ-th decomposition level.

Put $\hat{\mathbf{y}}[n]_\mu \stackrel{\text{def}}{=} \hat{y}^0_{[\mu]}[n]_\mu$, $n \in \mathbb{Z}$.

Compute the DFT $\{\hat{y}_0[n]_{\mu+1}\}$ and $\{\hat{y}_1[n]_{\mu+1}\}$ of the even and the odd polyphase components of the signal $\mathbf{y}$:

$$\hat{y}_0[n]_{\mu+1} = \frac{\hat{y}[n]_\mu + \hat{y}[n + N/2^{\mu+1}]_\mu}{2}, \quad \hat{y}_1[n]_1 = \frac{\hat{y}[n]_\mu - \hat{y}[n + N/2^{\mu+1}]_\mu}{2\omega^{-2^\mu k}}$$

Compute the IDFT

$$y^0_{[m]}[k] = \frac{2^m}{N} \sum_{n=0}^{N/2^m - 1} \omega^{2^m kn}\, \hat{y}^0_{[m]}[n]_m$$

and store the transform coefficients $\mathbf{y}^0_{[m]} \stackrel{\text{def}}{=} \left\{ y^0_{[m]}[k] \right\}$, $k = 0, \ldots, N/2^m - 1$

Algorithm 2 Inverse transform

3: Initialization: Compute the DFT

$$\hat{y}^s_{[m]}[n]_m = \sum_{k=0}^{N/2^m-1} \omega^{-2^m kn}\, y^s_{[m]}[k],\ k = 0, \ldots, N/2^m - 1, \quad s = 0, 1$$

For $\mu = m - 1$ to 0 Implement the one-level inverse transform

$$\begin{pmatrix} \hat{y}_0[n]_\mu \\ \hat{y}_1[n]_\mu \end{pmatrix} = \mathbf{P}_{[\mu]}[n] \cdot \begin{pmatrix} \hat{y}^0_{[\mu+1]}[n]_{\mu+1} \\ \hat{y}^1_{[\mu+1]}[n]_{\mu+1} \end{pmatrix}$$

Compute the DFT $\hat{y}^1_{[\mu]}[n]_\mu = \sum_{k=0}^{N/2^\mu-1} \omega^{-2^\mu kn}\, y^s_{[\mu]}[k]$, $n = 0, \ldots, N/2^\mu - 1$
Compute the DFT

$$\hat{y}^0_{[\mu]}[n]_\mu = \hat{y}_0[n]_{\mu+1} + \omega^{-2^\mu kn} \hat{y}_1[n]_{\mu+1}$$

Compute the IDFT

$$y_0[k] = \frac{2}{N} \sum_{n=0}^{N/2-1} \omega^{2kn}\, \hat{y}_0[n]_1, \quad y_1[k] = \frac{2}{N} \sum_{n=0}^{N/2-1} \omega^{2kn}\, \hat{y}_1[n]_1$$

Put $x[2k] = y_0[k]$, $x[2k+1] = y_1[k]$

Another implementation algorithm, which is based on the so-called *lifting scheme*, is presented in Sect. 16.1.

15.2 Compactly Supported Biorthogonal Wavelets

In this section we outline the design scheme of compactly supported (anti) symmetric wavelets, which is presented in [2]. The scheme is to be the adapted to the discrete-time periodic setting.

Section 15.1.3 describes how to derive a multilevel biorthogonal wavelet transform from biorthogonal two channel p-filter banks. It follows from Proposition 15.2 that, once we have a biorthogonal pair $\left\{\tilde{\mathbf{h}}^0,\ \mathbf{h}^0\right\}$ of the low-pass p-filters such that

$$\hat{h}^0[n]\,\hat{\tilde{h}}^0[-n] + \hat{h}^0[n + N/2]\,\hat{\tilde{h}}^0[-n + N/2] = 2, \tag{15.19}$$

the high-pass p-filters $\left\{\tilde{\mathbf{h}}^1,\ \mathbf{h}^1\right\}$ complementing the pair to the biorthogonal p-filter bank are derived straightforward. Certainly, if the p-filters $\left\{\tilde{\mathbf{h}}^0,\ \mathbf{h}^0\right\}$ and, consequently, the p-filters $\left\{\tilde{\mathbf{h}}^1,\ \mathbf{h}^1\right\}$, have finite impulse response up to periodicity (p-FIR) and linear phase, then the corresponding discrete-time wavelets of the first

discrete level are compactly supported (up to periodicity) and (anti-)symmetric. The wavelets of the subsequent decomposition levels are derived iteratively as described in Sect. 15.1.3. They also have compact supports whose span doubles when passing from a decomposition level to the next one. Thus, the key point is the design of an appropriate biorthogonal pair $\left\{\tilde{\mathbf{h}}^0,\ \mathbf{h}^0\right\}$ of the low-pass p-filters operating in the signal space $\Pi[N]$.

15.2.1 Biorthogonal Low-Pass FIR p-Filters

Start with a proposition, which is established in [2]. Define the polynomial of degree $p-1$

$$P_p(t) \stackrel{\text{def}}{=} \sum_{k=0}^{p-1} \binom{p-1+k}{k} t^k, \quad p = 2, 3 \ldots . \tag{15.20}$$

Proposition 15.4 [2] *There holds the identity*

$$(1-t)^p\ P_p(t) + t^p\ P_p(1-t) \equiv 1. \tag{15.21}$$

Take $t = \sin^2 \pi n/N$. Then, $1 - t = \cos^2 \pi n/N = \sin^2 \pi (n + N/2)/N$ and Eq. (15.21) entails the following trigonometric identity:

$$\cos^{2p} \frac{\pi n}{N}\ P\left(\sin^2 \frac{\pi n}{N}\right) + \cos^{2p} \frac{\pi (n+N/2)}{N}\ P\left(\sin^2 \frac{\pi (n+N/2)}{N}\right) \equiv 1. \tag{15.22}$$

Comparing Eq. (15.22) with the biorthogonality relation Eq. (15.19), we see that, once we factorize the trigonometric polynomial

$$T_p[n] \stackrel{\text{def}}{=} 2\cos^{2p} \frac{\pi n}{N}\ P_p[n]\left(\sin^2 \frac{\pi n}{N}\right) = Q\left(\sin^2 \frac{\pi n}{N}\right)\ \tilde{Q}\left(\sin^2 \frac{\pi n}{N}\right) \tag{15.23}$$

into a product of two polynomials of $\sin^2 \pi n/N$ such that $Q(0)\,\tilde{Q}(0) = 2$, then the polynomials Q and $\tilde{Q}$ can serve as the frequency responses of the biorthogonal low-pass FIR p-filters with linear phase:

$$\hat{h}^0[n] = Q\left(\sin^2 \frac{\pi n}{N}\right), \quad \hat{\tilde{h}}^0[n] = Q\left(\sin^2 \frac{\pi n}{N}\right). \tag{15.24}$$

The complementary high-pass p-filters can be produced in line with Eqs. (15.10) and (15.11).

An apparent way to factorize the polynomial, when the degree $p > 1$, is

$$\hat{h}^0[n] = q\cos^{2(p-s)} \frac{\pi n}{N}, \quad \hat{\tilde{h}}^0[n] = \frac{2}{q}\cos^{2p} \frac{\pi n}{N}\ P\left(\sin^2 \frac{\pi n}{N}\right), \quad q > 0, \tag{15.25}$$

where $s < p$ is a natural number. Equation (13.1.1) implies that $\cos^{2r} \pi n/N$ is the DFT of the discrete B-spline $\mathbf{b}^{2r}_{[1]}$, whose span is 2, and the order is $2r$. Thus, the impulse response of the synthesis low-pass p-filter $h[k] = b^{2r}_{[1]}[k]$.

An advantage of such a factorization is that the p-filters coefficients are rational numbers. Thus, filtering can be implemented in time domain via integer operations. Many examples of p-filters derived via factorization of Eq. (15.25) type are given in [2]. We discuss only one example of the so-called 5/3 p-filters, which are utilized in the JPEG 2000 standard for the lossless image compression.

15.2.2 Examples

15.2.2.1 5/3 p-Filters

Assume $p = 2$. Then,

$$T_2[n] = 2\cos^4 \frac{\pi n}{N} \left(1 + 2\sin^2 \frac{\pi n}{N}\right) = \left(2\cos^2 \frac{\pi n}{N}\right) \left(\cos^2 \frac{\pi n}{N} \left(1 + 2\sin^2 \frac{\pi n}{N}\right)\right).$$

Consequently, we define,

$$\begin{aligned} \hat{\tilde{h}}^0[n] &= 2\cos^2 \frac{\pi n}{N} = \frac{\omega^n}{2} + 1 + \frac{\omega^{-n}}{2}, \\ \hat{h}^0[n] &= \cos^2 \frac{\pi n}{N} \left(1 + 2\sin^2 \frac{\pi n}{N}\right) = \frac{1}{8} \left(\omega^n + 2 + \omega^{-n}\right) \left(4 - \omega^n - \omega^{-n}\right) \\ &= -\frac{\omega^{2n}}{8} + \frac{\omega^n}{4} + \frac{3}{4} + \frac{\omega^{-n}}{4} - \frac{\omega^{-2n}}{8}. \end{aligned} \tag{15.26}$$

Obviously, $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ are low-pass p-filters. The complementary high-pass p-filters are

$$\begin{aligned} \hat{h}^1[n] &= \omega^n \, \hat{\tilde{h}}^0[-n + N/2] = 2\omega^n \, \sin^2 \frac{\pi n}{N} = -\frac{\omega^{2n}}{2} + \omega^n - \frac{1}{2}, \\ \hat{\tilde{h}}^1[n] &= \omega^n \, \hat{h}^0[-n + N/2] = \omega^n \, \sin^2 \frac{\pi n}{N} \left(1 + 2\cos^2 \frac{\pi n}{N}\right) \\ &= \frac{\omega^n}{8} \left(-\omega^n + 2 - \omega^{-n}\right) \left(4 + \omega^n + \omega^{-n}\right) \\ &= -\frac{\omega^{3n}}{8} - \frac{\omega^{2n}}{4} + \frac{3\omega^n}{4} - \frac{1}{4} - \frac{\omega^{-n}}{8}. \end{aligned} \tag{15.27}$$

The impulse responses of the p-filters (up to periodicity), which are the discrete-time wavelets of the first decomposition level, are given in Table 15.1.

Table 15.1 Impulse responses of the 5/3 p-filters

k	−2	−1	0	1	2	3
$\tilde{h}^0[k] = \tilde{\psi}^0_{[1]}[k]$	–	1/2	1	1/2	–	–
$h^0[k] = \psi^0_{[1]}[k]$	−1/8	1/4	3/4	1/4	−1/8	–
$h^1[k] = \psi^1_{[1]}[k]$	–	–	−1/2	1	−1/2	–
$\tilde{h}^1[k] = \tilde{\psi}^1_{[1]}[k]$	–	−1/8	−1/4	3/4	−1/4	−1/8

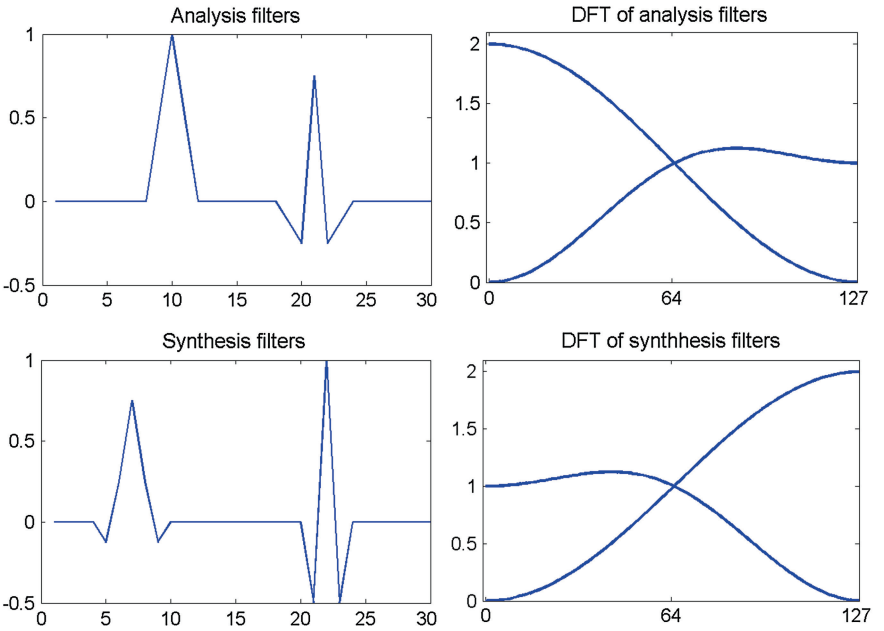


Fig. 15.2 Impulse and magnitude responses of the 5/3 p-filters

The analysis and synthesis wavelets of lower decomposition levels generated by the 5/3 p-filters are derived using Eq. (15.18). The impulse and magnitude responses of the 5/3 p-filters are displayed in Fig. 15.2.

15.2.2.2 9/7 p-Filters

Another way to factorize the trigonometric polynomial $T_p[n]$, which leads to synthesis and analysis p-filters, whose lengths are close to each other, is to factorize the polynomial $P_p(t)$ defined in Eq. (15.20): $P_p(t) = P_p^1(t)\, P_p^2(t)$, where $P_p^1(t)$ and $P_p^2(t)$ are real-valued polynomials, and to define

Table 15.2 Impulse responses of the low-pass 9/7 p-filters

k	$h[k] = \psi^0_{[1]}[k]$	$\tilde{h}[k] = \tilde{\psi}^0_{[1]}[k]$
0	0.85269867900889	0.78848561640637
± 1	0.37740285561283	0.41809227322204
± 2	−0.11062440441844	−0.04068941760920
± 3	−0.02384946501956	−0.06453888262876
± 4	0.03782845554969	0

$$T_p[n] \stackrel{\text{def}}{=} 2\cos^{2p}\frac{\pi n}{N}\, P_p[n]\left(\sin^2\frac{\pi n}{N}\right) = Q\left(\sin^2\frac{\pi n}{N}\right)\tilde{Q}\left(\sin^2\frac{\pi n}{N}\right) = \hat{\tilde{h}}^0[n]\,\hat{\tilde{h}}^0[-n]$$

$$Q(\left(\sin^2\frac{\pi n}{N}\right) = \sqrt{2}\cos^p\frac{\pi n}{N}\, P_p^1[n]\left(\sin^2\frac{\pi n}{N}\right),$$

$$\tilde{Q}(\left(\sin^2\frac{\pi n}{N}\right) = \sqrt{2}\cos^p\frac{\pi n}{N}\, P_p^2[n]\left(\sin^2\frac{\pi n}{N}\right).$$

If degrees of polynomials $P_p^1(t)$ and $P_p^2(t)$ differ by 1 then lengths of the impulse responses of the synthesis $\mathbf{h}^0$ and the analysis $\tilde{\mathbf{h}}^0$ p-filters differ by two. These biorthogonal p-filters are close to orthogonal ones. An important example is the set of so-called 9/7 p-filters, which are utilized in the JPEG 2000 standard for lossy image compression.

Assume that $p = 4$. Then,

$$T_4[n] = 2\cos^8\frac{\pi n}{N}\, P_4\left(\sin^2\frac{\pi n}{N}\right), \quad P_4(t) = 1 + 4\,t + 10\,t^2 + 20\,t^3.$$

The polynomial

$$P_4(t) = 20(t + a)\,(t^2 + b_1 t + b_0), \quad a \approx 0.342384094858369,$$
$$b_1 \approx 0.157615905141630, \quad b_0 \approx 0.146034820982800.$$

Define

$$\begin{aligned}
\hat{\tilde{h}}^0[n] &= \tilde{C}\,\cos^4\frac{\pi n}{N}\,\left(\sin^2\frac{\pi n}{N} + a\right) \\
&= \tilde{C}\,\frac{\left(\omega^n + 2 + \omega^{-n}\right)^2}{16}\,\left(a + \frac{-\omega^n + 2 - \omega^{-n}}{4}\right) \\
\hat{h}^0[n] &= C\,\cos^4\frac{\pi n}{N}\,\left(\sin^4\frac{\pi n}{N} + b_1\,\sin^2\frac{\pi n}{N} + b_0\right) \qquad (15.28) \\
&= C\,\frac{\left(\omega^n + 2 + \omega^{-n}\right)^2}{16}\,\left(b_0 + b_1\,\frac{-\omega^n + 2 - \omega^{-n}}{4} + \frac{(-\omega^n + 2 - \omega^{-n})^2}{16}\right), \\
\tilde{C} &\approx 1.032622122063015, \quad C = 2/\tilde{C} \approx 1.936816921958170.
\end{aligned}$$

The impulse responses of the low-pass 9/7 p-filters are given in Table 15.2.

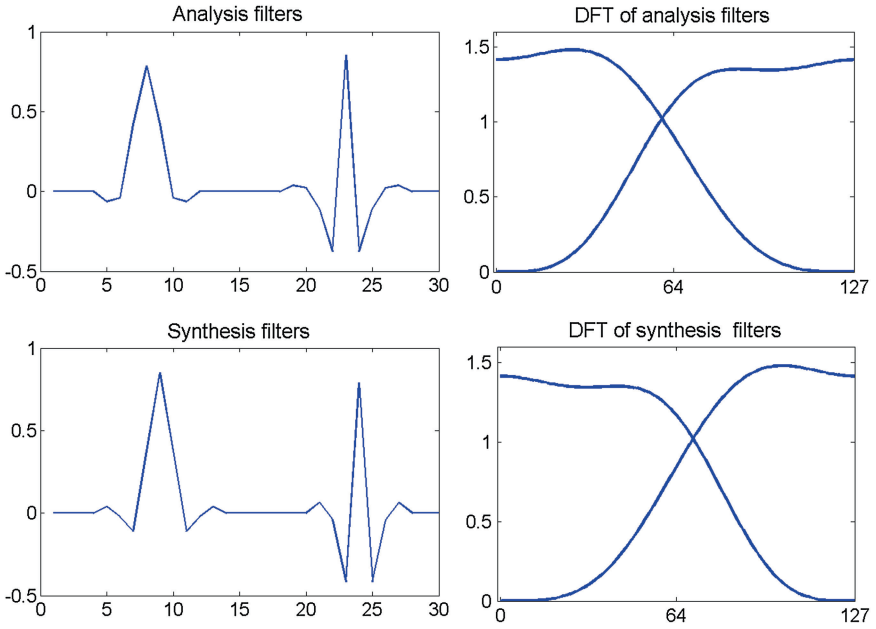


Fig. 15.3 Impulse and magnitude responses of the 9/7 p-filters

The frequency responses of the high-pass p-filters are

$$\begin{aligned}\hat{h}^1[n] &= \omega^n \tilde{C}\, \sin^4 \frac{\pi n}{N} \left(\cos^2 \frac{\pi n}{N} + a\right) \\ &= \omega^n \tilde{C}\, \frac{\left(-\omega^n + 2 - \omega^{-n}\right)^2}{16} \left(a + \frac{\omega^n + 2 + \omega^{-n}}{4}\right) \\ \hat{\tilde{h}}^1[n] &= \omega^n C\, \sin^4 \frac{\pi n}{N} \left(\cos^4 \frac{\pi n}{N} + b_1\, \cos^2 \frac{\pi n}{N} + b_0\right), \\ &= C\, \frac{\left(-\omega^n + 2 - \omega^{-n}\right)^2}{16} \left(b_0 + b_1\, \frac{\omega^n + 2 + \omega^{-n}}{4} + \frac{(\omega^n + 2 + \omega^{-n})^2}{16}\right).\end{aligned} \tag{15.29}$$

The impulse responses of the high-pass 9/7 p-filters:

$$h^1[k] = \psi^1_{[1]}[k] = (-1)^k \tilde{h}^0[-k-1], \quad \tilde{h}^1[k] = \tilde{\psi}^1_{[1]}[k] = (-1)^k h^0[-k-1], \ k \in \mathbb{Z}.$$

Remark 15.2.1 Filtering with the 9/7 p-filters can be implemented either directly in the time domain, utilizing the impulse responses, or in the frequency domain, in line with the algorithm described in Sect. 15.1.4. An alternative way to implement the transforms is based on the so-called *lifting scheme* to be discussed in Sect. 16.1.

The impulse and magnitude responses of the 9/7 p-filters are displayed in Fig. 15.3.

15.3 Restoration of Sampled Polynomials and Discrete Vanishing Moments

It is apparent that, as the frequency response of a low-pass p-filter **h** at the vicinity of zero becomes more flat, it better restores band-limited signals. To be specific, if $\hat{h}[n] \equiv c$ as $|n| \le n_0$ (up to a period N) and the DFT $\hat{x}[n]$ of a signal $\mathbf{x} \in \Pi[N]$ is zero outside the interval $-n_0 \le n \le n_0$, then application of the p-filter **h** exactly (up to the constant) restores the signal **x** by $\sum_{l=0}^{N-1} h[k-l]\, x[l] = c\, x[k]$.

15.3.1 Restoration of Sampled Polynomials

Flatness of the frequency response of a low-pass p-filter can be characterized by the difference $\hat{h}[n] - \hat{h}[0]$. Under the assumption that the frequency response $\hat{h}[n]$ is a rational function of $\omega^n = e^{2\pi i n/N}$, which has no poles while $n \in \mathbb{Z}$, the difference $\hat{h}[n] - \hat{h}[0] = (\omega^n - 1)^m\ \alpha[n],$, where m is some natural number, and $\alpha[n]$ is a rational function of ω^n, which has no poles while $n \in \mathbb{Z}$ and $\alpha[0] \neq 0$. The higher the multiplicity m of the root is, the flatter the frequency response.

Equation (1.26) claims that the sequence $(\omega^n - 1)^m$ is the DFT of the circular finite difference of order m and $(\omega^n - 1)^m \hat{x}[n] = \Delta^{\hat{m}}[\mathbf{x}][n]$. The sequence $\{\alpha[n]\}$ can be regarded as the frequency response of a low-(all-)pass p-filter **a**. Then, application of the p-filter **h** to a signal $\mathbf{x} \in \Pi[N]$ can be represented as

$$\mathbf{h}\,\mathbf{x} = \hat{h}[0]\,\mathbf{x} + \mathbf{a}\,\Delta^m[\mathbf{x}]. \tag{15.30}$$

Certainly, sampled polynomials do not belong to $\Pi[N]$ and p-filters can not be applied to them. However, the following facts related to the classical Fix-Strang condition [6] hold.

Proposition 15.5 *Assume that the frequency response of the low-pass p-filter* **h** *can be represented as*

$$\hat{h}[n] = \hat{h}[0] + \left(\omega^n - 1\right)^l\ \alpha[n] + \sum_{\mu=1}^{m-1} \left(\omega^n - 1\right)^\mu\ \beta_\mu, \tag{15.31}$$

where m is some natural number, $\alpha[n]$ is a rational function of ω^n, which has no poles for $n \in \mathbb{Z}$, and $\alpha[0] \neq 0$, and β_μ are constants. Suppose, **p** *is a signal from $\Pi[N]$, which coincides with a sampled polynomial $\mathbf{P}_{m-1}$ of degree $m-1$ at some interval: $p[k] = P_{m-1}(k)$ as $k = k_0, \ldots, k_m$, where $m < k_m - k_0 < N$. Then,*

$$\sum_{l=0}^{N-1} h[k-l]\, p[l] = Q_{m-1}(k),\ \text{as } k = k_0, \ldots, k_m - m - 1, \tag{15.32}$$

where $Q_{m-1}(t)$ is a polynomial of degree $m-1$. In a special case when all the coefficients $\beta_\mu = 0$

$$\sum_{l=0}^{N-1} h[k-l]\, p[l] = \hat{h}[0]\, P_{m-1}(k),\ \text{as } k = k_0, \ldots, k_m - m - 1. \tag{15.33}$$

Proof Under the condition Eq. (15.31) with all $\beta_\mu = 0$, application of the p-filter $\mathbf{h}$ to a signal $\mathbf{x} \in \Pi[N]$ can be represented as in Eq. (15.30). The finite difference of the signal $\mathbf{p}$

$$\Delta^m[\mathbf{p}][k] = \sum_{l=0}^{m} (-1)^l \binom{m}{l} p[k+l] = \Delta^m[\mathbf{P}_{m-1}](k),\ \text{as } k = k_0, \ldots, k_m - m - 1.$$

According to Proposition 1.1, $\Delta^m[\mathbf{P}_{m-1}][k] = 0$. Hence Eq. (15.33) follows. If some of the coefficients β_μ do not vanish then application of the p-filter $\mathbf{h}$ to the sampled polynomial P_{m-1} on the interval $k = k_0, \ldots, k_m$ results in

$$\sum_{l=0}^{N-1} h[k-l]\, p[l] = \hat{h}[0]\, P_{m-1}(k) + \sum_{\mu=1}^{m-1}, \beta_\mu\, \Delta^\mu[\mathbf{p}][k] = Q_{m-1}(k),$$

as $k = k_0, \ldots, k_m - m - 1$. ■

Definition 15.2 If Eq. (15.32) with a low-pass p-filter $\mathbf{h}$ holds then we say that the p-filter $\mathbf{h}$ locally reproduces sampled polynomials of degree $m-1$. If Eq. (15.33) is true then we say that the p-filter $\mathbf{h}$ locally restores sampled polynomials of degree $m-1$.

Proposition 15.6 *Assume that $R(z)$ is a rational function, which does not have a root at $z = 1$, $R(z) = P_\nu(z)/Q_\kappa(z)$ and the polynomials $P_\nu(z)$ and $Q_\kappa(z)$ are structured as $P_\nu(z) = p + (z-1)^s P_{\nu-s}(z)$, $Q_\kappa(z) = q + (z-1)^r Q_{\kappa-r}(z)$. If $p \neq 0$ then $R(z) = R(1) + (z-1)^{\min(s,r)} \tilde{R}(z)$, where $\tilde{R}(z)$ is a rational function, which does not have root at $z = 1$.*

The proof is straightforward.

15.3.2 Discrete Vanishing Moments

Proposition 15.7 *Assume that the frequency response of the high(band)-pass p-filter $\mathbf{g}$ can be represented as $\hat{g}[n] = (\omega^n - 1)^m\ \alpha[n]$, where m is some natural number and $\alpha[n]$ is a rational function of ω^n, which has no poles for $n \in \mathbb{Z}$ and $\alpha[0] \neq 0$. Assume $\mathbf{p}$ is a signal from $\Pi[N]$, and it coincides with a sampled polynomial $\mathbf{P}_{m-1}$ of degree $m-1$ at some interval $p[k] = P_{m-1}(k)$ as $k = k_0, \ldots, k_m$, where $m < k_m - k_0 < N$. Then, $\sum_{l=0}^{N-1} g[k-l]\, p[l] = 0$, as $k = k_0, \ldots, k_m - m - 1$.*

The proof is similar to the proof of Proposition 15.5.

Definition 15.3 If a high (band)-pass p-filter **g** satisfies the conditions of Proposition 15.7, we say that the p-filter **g** locally eliminates sampled polynomials of degree $m-1$. If a wavelet $\psi^1_{[1]}[k] \stackrel{\text{def}}{=} g[k],\ k \in \mathbb{Z}$, we say that the wavelet $\psi_{[1]}$ has m local discrete vanishing moments (LDVM).

Assume that **g** is a high-pass p-filter whose frequency response $\hat{g}[n]$ is a rational function of ω^n, which has no poles for $n \in \mathbb{Z}$ and $\hat{g}[N/2] \neq 0$. Then, in the vicinity of $N/2$, the frequency response can be represented as $\hat{g}[n] = \hat{g}[N/2] + (\omega^n + 1)^l\ \beta[n]$, where l is some natural number and $\beta[n]$ is a rational function of ω^n, which has no poles for $n \in \mathbb{Z}$ and $\beta[N/2] \neq 0$. The power l determines the flatness of the frequency response $\hat{g}[n]$ in the vicinity of $N/2$.

Remark 15.3.1 The statements of Propositions 15.5 and 15.7 concerning the low-pass **h** and high-pass **g** p-filters, respectively, remain true if the frequency responses are represented as

$$\hat{h}[n] = \hat{h}[0] + \sin^m \frac{\pi n}{N}\ \bar{\alpha}[n], \quad \hat{g}[n] = \sin^m \frac{\pi n}{N}\ \bar{\alpha}[n],$$

where m is some natural number and $\bar{\alpha}[n]$ is a rational function of ω^n, which has no poles for $n \in \mathbb{Z}$ and $\bar{\alpha}[0] \neq 0$.

Restoration of polynomials by low-pass filters coupled with their elimination by respective high–pass filters constituting the filter banks is important in signal processing applications. For example, when data compression is implemented by multiscale wavelet transforms, this property enables us to condense the information on the smooth component of a signal or an image into a small number of low-frequency transform coefficients.

15.3.3 Examples

5/3 wavelets The frequency responses of the low-pass p-filters are

$$\hat{\tilde{h}}^0[n] = 2\cos^2 \frac{\pi n}{N} = 2 - 2\sin^2 \frac{\pi n}{N} \tag{15.34}$$

$$\hat{h}^0[n] = \cos^2 \frac{\pi n}{N}\ \left(1 + 2\sin^2 \frac{\pi n}{N}\right) = 1 + \sin^2 \frac{\pi n}{N}\ \cos \frac{2\pi n}{N}. \tag{15.35}$$

Thus, the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore first degree polynomials.
The frequency responses of the high-pass p-filters are

$$\hat{h}^1[n] = 2\omega^n\ \sin^2 \frac{\pi n}{N} \quad \hat{\tilde{h}}^1[n] = \omega^n\ \sin^2 \frac{\pi n}{N}\ \left(1 + 2\cos^2 \frac{\pi n}{N}\right). \tag{15.36}$$

Therefore, the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ locally eliminate first degree polynomials. The wavelets $\psi^1_{[1]}[k] = h^1[k]$ and $\tilde{\psi}^1_{[1]}[k] = \tilde{h}^1[k]$ have two LDVM each.

9/7 wavelets Equations (15.28) imply that

$$\begin{aligned}
\hat{\tilde{h}}^0[n] - \hat{\tilde{h}}^0[0] &= \tilde{C}\left(\cos^4\frac{\pi n}{N}\left(\sin^2\frac{\pi n}{N} + a\right) - a\right) \\
&= \tilde{C}\,\sin^2\frac{\pi n}{N}\left(\cos^4\frac{\pi n}{N} - a\left(\cos^2\frac{\pi n}{N} + 1\right)\right), \\
\hat{h}^0[n] - \hat{h}^0[0] &= C\left(\cos^4\frac{\pi n}{N}\left(\sin^4\frac{\pi n}{N} + b_1\sin^2\frac{\pi n}{N} + b_0\right) - b_0\right) \\
&= C\,\sin^2\frac{\pi n}{N}\left(\cos^4\frac{\pi n}{N}\left(\sin^2\frac{\pi n}{N} + b_1\right) - b_0\left(\cos^2\frac{\pi n}{N} + 1\right)\right).
\end{aligned}$$

The 9/7 p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore first degree polynomials but locally reproduces cubic polynomials.

However, it is seen from Eq. (15.29) that the frequency responses of the high-pass p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ comprise the factor $\sin^4 \pi n/N$. Therefore, the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ locally eliminate cubic polynomials. The wavelets $\psi^1_{[1]}[k] = h^1[k]$ and $\tilde{\psi}^1_{[1]}[k] = \tilde{h}^1[k]$ have four LDVM each.

References

1. M.J. Smith, T.R. Barnwell, Exact reconstruction techniques for tree-structured subband coders. IEEE Trans. Acoust. Speech Signal Process. **34**(3), 434–441 (1986)
2. A. Cohen, I. Daubechies, J.-C. Feauveau, Biorthogonal bases of compactly supported wavelets. Comm. Pure Appl. Math. **45**(5), 485–560 (1992)
3. M. Vetterli, C. Herley, Wavelets and filter banks: Theory and design. IEEE Trans. Signal Process. **40**(9), 2207–2232 (1992)
4. M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using wavelet transform. IEEE Trans. Image Process. **1**(2), 205–220 (1992)
5. D.S. Taubman, M.W. Marcellin (eds.), *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, The Springer International Series in Engineering and Computer Science, vol. 642 (Springer, Berlin, 2002)
6. G. Strang, G. Fix, A Fourier analysis of the finite element variational method, in *Constructive Aspects of Functional Analysis*, ed. by G. Geymonat (Springer, Berlin, 2011), pp. 793–840

Chapter 16
Biorthogonal Wavelet Transforms Originating from Splines

Abstract This section describes how to generate families of biorthogonal wavelet transforms in spaces of periodic signals using prediction p-filters originating from polynomial and discrete splines. The wavelets related to those transforms are (anti)symmetric, well localized in time domain and have flat spectra. The families contain low-pass p-filters, which locally restore sampled polynomials of any degree, while the respective high-pass p-filters locally eliminate polynomials of the same degrees.

16.1 Lifting Scheme of Wavelet Transforms

A useful tool for design and implementation of biorthogonal wavelet transforms is provided by the so-called *lifting scheme* introduced by Sweldens [8–10].

16.1.1 Lifting Steps

Generally, the lifting scheme of the wavelet transform of a signal consists of four steps: 1. Split, 2. Predict, 3. Update (lifting), and 4. Normalization. To reconstruct the signal from the transform coefficients, the lifting steps are reversed.

16.1.1.1 Decomposition

The following are the lifting steps for the decomposition of a signal $\mathbf{x} \in \Pi[N]$.

Split: The signal $\mathbf{x} = \{x[l]\} \in \Pi[N]$ is split into its polyphase components: $\mathbf{x} = \mathbf{x}_0 + \mathbf{x}_1$, where $\mathbf{x}_p = \{x_p[l] \stackrel{\text{def}}{=} x[2l + p]\}_{l \in \mathbb{Z}}$, $p = 0, 1$.

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_16,

Predict: The even polyphase component is filtered by some low-pass *prediction* p-filter $\tilde{\mathbf{f}}$ in order for the filtered version of the even component $\mathbf{x}_0$ to predict the odd component $\mathbf{x}_1$. Then, the existing odd array $\mathbf{x}_1$ is replaced by the array $\mathbf{a}^1$, which is the difference between $\mathbf{x}_1$ and the predicted array.
In the frequency domain, the operations are described as

$$\hat{a}^1[n]_1 = \hat{x}_1[n]_1 - \hat{\tilde{f}}[n]_1\, \hat{x}_0[n]_1$$

Update (Lifting): Generally, downsampling the original signal $\mathbf{x}$ into $\mathbf{x}_0$ depletes the signal's smoothness. To obtain a sparse signal similar to the original $\mathbf{x}$, and, by this means, to eliminate the spectral aliasing, the produced array $\mathbf{a}^1$ is filtered by a low-pass *update* filter, which we prefer to denote by $\mathbf{f}/2$. The filtered array is used to update the even array $\mathbf{x}_0$ resulting in the array $\mathbf{a}^0$:

$$\hat{a}^0[n]_1 = \hat{x}_0[n]_1 + \frac{1}{2}\hat{f}[n]_1\, \hat{a}^1[n]_1.$$

Provided that the p-filter $\mathbf{f}$ is properly chosen, the even array $\mathbf{x}_0$ is transformed into a smoothed and downsampled replica of $\mathbf{x}$.

Normalization: Finally, the smoothed array $\mathbf{y}^0_{[1]}$ and the array of details $\mathbf{y}^1_{[1]}$ are obtained by the operation $\mathbf{y}^0_{[1]} = \sqrt{2}\,\mathbf{a}^0, \quad \mathbf{y}^1_{[1]} = \mathbf{a}^1/\sqrt{2}$.

16.1.1.2 Reconstruction

One of the most attractive features of lifting schemes is that reconstruction of the signal $\mathbf{x}$ from the arrays $\mathbf{y}^0_{[1]}$ and $\mathbf{y}^1_{[1]}$ is implemented by reverse decomposition:

Undo Normalization: $\mathbf{a}^0 = \mathbf{y}^0_{[1]}/\sqrt{2} \quad \mathbf{a}^1 = \sqrt{2}\,\mathbf{y}^1_{[1]}$.

Undo Lifting: The even polyphase component $\hat{x}_0[n]_1 = \hat{a}^0[n]_1 - \frac{1}{2}\hat{f}[n]_1\, \hat{a}^1[n]_1$ is restored.

Undo Predict: The odd polyphase component $\hat{x}_1[n]_1 = \hat{a}^1[n]_1 + \hat{\tilde{f}}[n]_1\, \hat{x}_0[n]_1$ is restored.

Undo Split: The last step is the standard restoration of the signal from its even and odd components. In the frequency domain, it appears as $\hat{x}[n] = \hat{x}_0[n]_1 + \omega^{-n}\, \hat{x}_1[n]_1$.

16.1.2 Filter Banks

Rewriting the lifting steps in a matrix form, we obtain the polyphase matrices of the wavelet transforms:

$$\begin{aligned}\begin{pmatrix}\hat{y}^0[n]_1\\ \hat{y}^1[n]_1\end{pmatrix} &= \begin{pmatrix}\sqrt{2} & 0\\ 0 & 1/\sqrt{2}\end{pmatrix}\cdot\begin{pmatrix}1 & \hat{f}[n]_1/2\\ 0 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & 0\\ -\hat{\tilde{f}}[n]_1 & 1\end{pmatrix}\cdot\begin{pmatrix}\hat{x}_0[n]_1\\ \hat{x}_1[n]_1\end{pmatrix}\\ &= \tilde{\mathbf{P}}[-n]\cdot\begin{pmatrix}\hat{x}_0[n]_1\\ \hat{x}_1[n]_1\end{pmatrix}.\end{aligned}$$

Hence, the analysis polyphase matrix is

$$\tilde{\mathbf{P}}[n] = \begin{pmatrix}\sqrt{2}\left(1-\hat{\tilde{f}}[-n]_1\hat{f}[-n]_1/2\right) & \hat{f}[-n]_1/\sqrt{2}\\ -\hat{\tilde{f}}[-n]_1/\sqrt{2} & 1/\sqrt{2}\end{pmatrix}.$$

The reconstruction operations are represented by

$$\begin{aligned}\begin{pmatrix}\hat{x}_0[n]_1\\ \hat{x}_1[n]_1\end{pmatrix} &= \begin{pmatrix}1 & 0\\ \hat{\tilde{f}}[n]_1 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & -\hat{f}[n]_1/2\\ 0 & 1\end{pmatrix}\cdot\begin{pmatrix}1/\sqrt{2} & 0\\ 0 & \sqrt{2}\end{pmatrix}\cdot\begin{pmatrix}\hat{y}^0[n]_1\\ \hat{y}^1[n]_1\end{pmatrix}\\ &= \mathbf{P}[n]\cdot\begin{pmatrix}\hat{y}^0[n]_1\\ \hat{y}^1[n]_1\end{pmatrix}.\end{aligned}$$

Hence, the synthesis polyphase matrix is

$$\mathbf{P}[n] = \begin{pmatrix}1/\sqrt{2} & -\hat{f}[n]_1/\sqrt{2}\\ \hat{\tilde{f}}[n]_1/\sqrt{2} & \sqrt{2}\left(1-\hat{\tilde{f}}[n]_1\hat{f}[n]_1/2\right)\end{pmatrix}.$$

Obviously, the perfect reconstruction condition $\mathbf{P}[n]\cdot\tilde{\mathbf{P}}[-n] = \mathbf{I}_2$ is satisfied. Recall that the polyphase matrices of p-filter banks $\left\{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1\right\}$ and $\left\{\mathbf{h}^0, \mathbf{h}^1\right\}$ are, respectively

$$\tilde{\mathbf{P}}[n] \overset{\text{def}}{=} \begin{pmatrix}\hat{\tilde{h}}_0^0[n]_1 & \hat{\tilde{h}}_1^0[n]_1\\ \hat{\tilde{h}}_0^1[n]_1 & \hat{\tilde{h}}_1^1[n]_1\end{pmatrix}, \quad \mathbf{P}[n] \overset{\text{def}}{=} \begin{pmatrix}\hat{h}_0^0[n]_1 & \hat{h}_0^1[n]_1\\ \hat{h}_1^0[n]_1 & \hat{h}_1^1[n]_1\end{pmatrix}.$$

Thus, the frequency responses of the synthesis p-filters are

$$\begin{aligned}\hat{h}^0[n] &= \frac{1}{\sqrt{2}}\left(1+\omega^{-n}\hat{\tilde{f}}[n]_1\right)\\ \hat{h}^1[n] &= \omega^{-n}\sqrt{2}\left(1-\frac{1}{2}\left(\hat{\tilde{f}}[n]_1\hat{f}[n]_1-\omega^n\hat{f}[n]_1\right)\right)\\ &= \frac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^n\hat{f}[n]_1+W[n]_1\right),\end{aligned} \tag{16.1}$$

where

$$W[n]_1 \overset{\text{def}}{=} 1-\hat{\tilde{f}}[n]_1\hat{f}[n]_1. \tag{16.2}$$

The frequency responses of the analysis p-filters are

$$\begin{aligned}\hat{\tilde{h}}^0[n] &= \tfrac{1}{\sqrt{2}}\left(1+\omega^{-n}\hat{f}[-n]_1 + W[-n]_1\right) = \omega^{-n}\,\hat{h}^1[-n+N/2],\\ \hat{\tilde{h}}^1[n] &= \tfrac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^{n}\hat{\tilde{f}}[-n]_1\right) = \omega^{-n}\,\hat{h}^0[-n+N/2].\end{aligned} \tag{16.3}$$

The pairs $\left\{\tilde{\mathbf{h}}^0,\ \tilde{\mathbf{h}}^1\right\}$ and $\left\{\mathbf{h}^0,\ \mathbf{h}^1\right\}$ constitute a perfect reconstruction biorthogonal p-filter bank. Thus, the transform of a signal $\mathbf{x} \longrightarrow \mathbf{y}^0 \bigcup \mathbf{y}^1$ can be regarded as one step of a wavelet transform, where $\tilde{\psi}^0_{[1]} \stackrel{\text{def}}{=} \tilde{\mathbf{h}}^0$ and $\tilde{\psi}^1_{[1]} \stackrel{\text{def}}{=} \tilde{\mathbf{h}}^0$ are the analysis discrete-time wavelets, while $\psi^0_{[1]} \stackrel{\text{def}}{=} \mathbf{h}^0$ and $\psi_{[1]} \stackrel{\text{def}}{=} \mathbf{h}^0$ are the synthesis discrete-time wavelets. The analysis and synthesis wavelets are interchangeable.

If the DFT of the zero polyphase component of a p-filter $\mathbf{h}$ is a constant $\hat{h}_0[n]_1 = c$ then the p-filter is regarded as interpolating. We see from (16.1) that the low-pass synthesis p-filter $\mathbf{h}^0$ is interpolating. Consequently, the zero polyphase component of the synthesis wavelet $\psi^0_{[1]}$ interpolates the Kroneker delta (up to the constant $c = 2^{-1/2}$). It means that

$$\psi^0_{[1]}[2l] = \begin{cases} c, & \text{if } l = kN/2,\ k \in \mathbb{Z};\\ 0, & \text{otherwise.}\end{cases} \qquad l \in \mathbb{Z}.$$

Properties of the p-filter banks and the corresponding wavelets are determined by choice of the predict $\tilde{\mathbf{f}}$ and the update $\mathbf{f}$ p-filters. The perfect reconstruction and the biorthogonality conditions for the presented p-filter banks are satisfied with any choice of low-pass p-filters $\tilde{\mathbf{f}}$ and $\mathbf{f}$. However, utilization of polynomial and discrete splines for the p-filters design produces a diverse family of biorthogonal wavelets. Properties of those wavelets such as symmetry, interpolation, smoothness, flat spectra, good time-domain localizations and vanishing moments fit well signal processing needs. Implementation of these transforms is highly efficient.

Remark 16.1.1 *Once a predict p-filter $\tilde{\mathbf{f}}$ is selected, any low-pass p-filter can serve as the update p-filter $\mathbf{f}$. For simplicity, in what follows, we choose the update p-filter $\mathbf{f}$ to be equal to the time-reversed predict p-filter $\tilde{\mathbf{f}}$, that is, $\hat{f}[n]_1 = \hat{\tilde{f}}[-n]_1$. Other options are discussed in [1–3].*

Proposition 16.1 *If the update p-filter frequency response $\hat{f}[n]_1 = \hat{\tilde{f}}[-n]_1$ and the symmetry property $\omega^{-n}\,\hat{f}[-n]_1 = \omega^{n}\,\hat{f}[n]_1$ holds then the frequency responses of the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^1$ are*

$$\hat{h}^0[n] = \frac{1}{\sqrt{2}}\left(1+\omega^{-n}\hat{f}[-n]_1\right), \quad \hat{\tilde{h}}^1[n] = \frac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^{n}\hat{f}[n]_1\right), \tag{16.4}$$

and the sequence $W[n]_1$ can be represented in the following way:

$$W[n]_1 = 1 - |\hat{f}[n]_1|^2 = (1 - \omega^{-n}\hat{f}[n]_1)\,(1 + \omega^n \hat{f}[-n]_1) \qquad (16.5)$$
$$= 2\omega^n \hat{h}^0[n]\,\hat{\tilde{h}}^1[n] = W[-n]_1.$$

Proof The symmetry property implies Eq. (16.4), and the relation $W[n]_1 = 1 - |\hat{f}[n]_1|^2$. On the other hand,

$$2\omega^n \hat{h}^0[n]\,\hat{\tilde{h}}^1[n] = (1 - \omega^n \hat{f}[n]_1)\,(1 + \omega^{-n}\hat{f}[-n]_1)$$
$$= 1 - \omega^n \hat{f}[n]_1 + \omega^{-n}\hat{f}[-n]_1 - |\hat{f}[n]_1|^2$$
$$= 1 - \omega^n \hat{f}[-n]_1 + \omega^n \hat{f}[-n]_1 - |\hat{f}[n]_1|^2 = 1 - |\hat{f}[n]_1|^2 = W[n]_1.$$

■

Corollary 16.1 *Under conditions of Proposition 16.1, the frequency response of the p-filters* $\mathbf{h}^1$ *and* $\tilde{\mathbf{h}}^0$ *are represented in the following way:*

$$\hat{\tilde{h}}^0[n] = \tfrac{1}{\sqrt{2}}\left(1 + \omega^n \hat{f}[n]_1 + W[n]_1\right) = \hat{h}^0[n]\left(1 + \sqrt{2}\,\omega^n \hat{\tilde{h}}^1[n]\right),$$
$$\hat{h}^1[n] = \tfrac{\omega^{-n}}{\sqrt{2}}\left(1 - \omega^n \hat{f}[n]_1 + W[n]_1\right) = \hat{\tilde{h}}^1[n]\left(1 + \sqrt{2}\,\omega^n \hat{h}^0[n]\right). \qquad (16.6)$$

The wavelet transform is extended to the second decomposition level by application of the lifting operations to the coefficients array $\mathbf{y}^0_{[1]} \stackrel{\text{def}}{=} \mathbf{y}^0$ using prediction and update p-filters $\tilde{\mathbf{f}}_{[1]}$ and $\mathbf{f}_{[1]}$, respectively, which operate in the subspace $\pi[N/2]$. The simplest way to derive the p-filters $\tilde{\mathbf{f}}_{[1]}$ and $\mathbf{f}_{[1]}$ is to define their frequency responses as $\hat{\tilde{f}}_{[1]}[n]_1 \stackrel{\text{def}}{=} \hat{\tilde{f}}[2n]_2$ and $\hat{f}_{[1]}[n]_2 \stackrel{\text{def}}{=} \hat{f}[2n]_1$. Extension of the transforms to the subsequent decomposition levels is similar. Reconstruction is implemented in an inverse order.

16.2 Prediction Filters Derived from Polynomial Splines

Splines provide flexible tools for design of the predict p-filters. The idea behind the design is to construct either a polynomial or a discrete spline, which interpolates or quasi-interpolates even samples of a signal, and to predict odd samples by the spline values at the midpoints between the (quasi-)interpolation points. In this section the polynomial splines case is discussed.

We deal with splines that are defined on the uniform grid $\{k\}$ and are periodic with the period $N/2 = 2^{j-1}$.

16.2.1 Periodic Interpolating Splines

The space of $N/2$-periodic splines of order p defined on the uniform grid $\{k\}$ is denoted by ${}^{p}\mathcal{S}_{[1]}$. A basis in this space is constituted by shifts of the $N/2$-periodic B-spline, which is denoted by $[B^{p}_{[1]}(t)$. Its Fourier series expansion is

$$B^{p}_{[1]}(t) = \frac{2}{N} \sum_{n \in \mathbb{Z}} e^{4\pi i n t/N} \left(\frac{\sin 2\pi n/N}{2\pi n/N} \right)^{p}.$$

The B-spline $B^{p}_{[1]}(t)$ is supported on the interval $(-p/2,\ p/2)$ (up to periodization).

Any spline $S^{p}(t)$ from ${}^{p}\mathcal{S}_{[1]}$ is represented as

$$S^{p}(t) = \sum_{k=0}^{N/2-1} q[k]\, B^{p}_{[1]}(t-k).$$

Its values at integer and semi-integer points are the circular convolutions

$$s_0[l] \stackrel{\text{def}}{=} S^{p}(l) = \sum_{k=0}^{N/2-1} q[k]\, b_0[l-k], \quad s_1[l] \stackrel{\text{def}}{=} S^{p}\left(l+\frac{1}{2}\right) = \sum_{k=0}^{N/2-1} q[k]\, b_1[l-k],$$

$$b_0[l] \stackrel{\text{def}}{=} B^{p}_{[1]}(l), \quad b_1[l] \stackrel{\text{def}}{=} B^{p}_{[1]}\left(l+\frac{1}{2}\right).$$

The DFT of the sequence $\{b_r[l]\}$, $r = 0, 1$, is denoted as

$$u^{p}_{[1]}[n] \stackrel{\text{def}}{=} \hat{b}_0[n]_1 = \sum_{l=0}^{N/2-1} \omega^{-2nl}\, b_0[l], \quad v^{p}_{[1]}[n] \stackrel{\text{def}}{=} \hat{b}_1[n]_1.$$

If the spline $S^{p}(t)$ interpolates the even polyphase component $\mathbf{x}_0 = \{x[2l]\}$ of a signal $\mathbf{x} \in \Pi[N]$ then its coefficients $q[k]$ can be explicitly calculated via the DFT

$$\hat{s}_0[n]_1 = \hat{q}[n]_1\, u^{p}_{[1]}[n] = \hat{x}_0[n]_1 \Longrightarrow \hat{q}[n]_1 = \frac{\hat{x}_0[n]_1}{u^{p}_{[1]}[n]}$$

$$\Longrightarrow \hat{x}_1[n]_1 = \hat{q}[n]_1\, v^{p}_{[1]}[n] = \frac{v^{p}_{[1]}[n]}{u^{p}_{[1]}[n]}\, \hat{x}_0[n]_1. \tag{16.7}$$

16.2.2 Prediction p-filters

Equation (16.7) means that, in order to predict the odd polyphase component $\mathbf{x}_1$ of a signal $\mathbf{x} \in \mathit{\Pi}[N]$ by the midpoint values of the spline, which interpolates the even polyphase component $\mathbf{x}_0$, one should filter the array $\mathbf{x}_0$ by the p-filter $\mathbf{f}_c^p$, whose frequency response is

$$\hat{f}_c^p[n]_1 \stackrel{\text{def}}{=} \frac{v_{[1]}^p[n]}{u_{[1]}^p[n]}. \tag{16.8}$$

Proposition 16.2 *The p-filter* $\mathbf{f}_c^p$ *is low-pass, and the following symmetry property holds:*

$$\omega^{-n}\, \hat{f}_c^p[n]_1 = \omega^n\, \hat{f}_c^p[-n]_1 \tag{16.9}$$

Proof The $N/2$-periodic sequence $u_{[1]}^p[n]$, which is the characteristic sequence of the spline space ${}^p\mathscr{S}_{[1]}$, is strictly positive. Due to symmetry and positiveness of the B-spline, $u_{[1]}^p[n] = u_{[1]}^p[-n]_1$, thus it is a cosine polynomial with positive coefficients. It is symmetric about $N/4$, where it has its single minimum. The maximal value of $u_{[1]}^p[n]$ is 1, being reached when $n = 0$. A few characteristic sequences are displayed in Fig. 3.4. As for the sequence $v_{[1]}^p[n]$, its value when $n = 0$ is $v_{[1]}^p[0] = \sum_{k=0}^{N/2-1} B_{[1]}^p(k + 1/2) > 0$ (indeed, $v_{[1]}^p[0] = 1$ for any natural p). Thus, $\hat{f}_c^p[0]_1 > 0$ and, consequently, $\mathbf{f}_c^p$ is low-pass p-filter. Define, the N-periodic sequence $\left\{b[l] \stackrel{\text{def}}{=} B_{[1]}^p(l/2)\right\}$. This sequence is symmetric about zero. Therefore, its DFT

$$\hat{b}[n] = \sum_{l=0}^{N-1} \omega^{-nl} b[l] = \hat{b}_0[n]_1 + \omega^{-n}\, \hat{b}_1[n]_1 = u_{[1]}^p[n] + \omega^{-n}\, v_{[1]}^p[n]$$

is symmetric about swap $n \to -n$. Since $u_{[1]}^p[n] = u_{[1]}^p[-n]_1$, the same is true for $\omega^{-n}\, v_{[1]}^p[n]$. Hence, Eq. (16.9) follows. ■

Table 16.1 presents the sequences $\{B^p(l)\}$ and $\{B^p(l + 1/2)\}$ for different values of p. The samples of higher order splines are calculated by the MATLAB function `bsuvP.m`. They are gathered in the file `BSUV.mat`.

The continuous counterparts of the sequences $u_{[1]}^p[n]$ and $v^p[n]_1$, which are discrete-time Fourier transforms of the non-periodic B-splines, were introduced and studied in [7]. Some additional properties of the sequences $u_{[1]}^p[n]$ and $v^p[n]_1$ are established in [3, 13].

Table 16.1 Sampled B-splines $\left\{B^p_{[1]}(l)\right\}$ and $\left\{B^p_{[1]}(l+1/2)\right\}$

l	-4	-3	-2	-1	0	1	2	3	4
$B^2(l)$	0	0	0	0	1	0	0	0	0
$B^3(l) \times 8$	0	0	0	1	6	1	0	0	0
$B^4(l) \times 6$	0	0	0	1	4	1	0	0	0
$B^5(l) \times 384$	0	0	1	76	230	76	1	0	0
$B^6(l) \times 120$	0	0	1	26	66	26	1	0	0
$B^7(l) \times 46080$	0	1	722	10543	23548	10543	722	1	0
$B^2(l+1/2) \times 2$	0	0	0	1	1	0	0	0	0
$B^3(l+1/2) \times 2$	0	0	0	1	1	0	0	0	0
$B^4(l+1/2) \times 48$	0	0	1	23	23	1	0	0	0
$B^5(l+1/2) \times 24$	0	0	1	11	11	1	0	0	0
$B^6(l+1/2) \times 3840$	0	1	237	1682	1682	237	1	0	0
$B^7(l+1/2) \times 720$	0	1	57	302	302	57	1	0	0

16.2.3 Approximation Properties of Spline p-filters

Approximation properties of interpolating splines are well investigated. In particular, the non-periodic interpolating spline of order p, which consists of piece-wise polynomials of degree $p-1$, restores polynomials of degree $p-1$. It means that a spline of order p, which interpolates a polynomial $\pi_{p-1}(t)$ of degree $p-1$ on the grid $\{l\}$, coincides with this polynomial: $S^p_{\pi_{p-1}}(t) \equiv \pi_{p-1}(t)$. In particular, $S^p_{\pi_{p-1}}(l+1/2) = \pi_{p-1}(l+1/2)$. However, when the spline order $p = 2r-1$ is an odd number, the so-called super-convergence property holds, that is, the approximation order of such splines at the midpoints between the interpolation points is higher than at the rest of points.

Specifically, *If a spline of order* $p = 2r-1$ *(degree* $p-1$*) interpolates a polynomial}* $\pi_p(t)$ *of degree pon the grid* $\{l\}$, *then* $S^p_{\pi_p}(l+1/2) = \pi_p(l+1/2)$. The super-convergence property was established in [13]. For periodic splines, the approximation properties hold locally [3], attested by the following fact:

Proposition 16.3 ([3]) *If the spline order is* $p = 2r$ *or* $p = 2r-1$ *then*

$$\omega^{-n}\,\hat{f}^p_c[n]_1 = 1 + \sin^{2r}\frac{\pi n}{N}\,\chi^{\,p}[n], \tag{16.10}$$

where r is a natural number and $\chi^{\,p}[n]$ *is a rational function of* ω^n*, which has no poles for* $n \in \mathbb{Z}$ *and no root at* $n = 0$.

The frequency responses $\hat{f}^p_c[n]_1$ of the p-filters $\mathbf{f}^p_c$, which are defined in Eq. (16.8), are readily calculated via the DFT of the sampled B-splines.

Therefore, the choice of interpolating splines, especially of the odd order splines, as a source for design of the prediction filters is well justified.

16.2.4 Perfect Reconstruction p-filter Banks

In order to design PR p-filter banks, which implement the biorthogonal wavelet transforms, the prediction p-filters are chosen to be $\tilde{\mathbf{f}} = \mathbf{f}_c^p$. Their frequency responses $\hat{f}_c^p[n]_1$ are defined in Eq. (16.8). The update p-filters $\mathbf{f}$ are equal to the time-reversed predict p-filters and, consequently, the frequency response $\hat{f}[n] = \hat{f}_c^p[-n]_1$.

Then, the sequence

$$W^p[n]_1 \stackrel{\text{def}}{=} 1 - \left|\hat{f}_c^p[n]_1\right|^2 = W^p[-n]_1. \tag{16.11}$$

Equations (16.4) and (16.6) imply that the frequency responses of the synthesis and the analysis p-filter banks derived from the interpolating spline of order p are

$$\begin{aligned} &\hat{h}^0[n] = \tfrac{1}{\sqrt{2}}\left(1 + \omega^{-n}\,\hat{f}_c^p[n]_1)\right), \quad \hat{\tilde{h}}^1[n] = \tfrac{\omega^{-n}}{\sqrt{2}}\left(1 - \omega^n\,\hat{f}_c^p[-n]_1\right), \\ &\hat{\tilde{h}}^0[n] = \hat{h}^0[n]\left(1 + \sqrt{2}\,\omega^n\,\hat{\tilde{h}}^1[n]\right), \quad \hat{h}^1[n] = \hat{\tilde{h}}^1[n]\left(1 + \sqrt{2}\,\omega^n \hat{h}^0[n]\right). \end{aligned} \tag{16.12}$$

Such a structure of the p-filters determines local restoration of sampled polynomials by the low-pass p-filters and their elimination by the high-pass ones. The following proposition stems from Proposition 16.3:

Proposition 16.4 *Assume that the spline order is either $p = 2r$ or $p = 2r - 1$. The low-pass p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ derived from the prediction p-filter $\tilde{\mathbf{f}} = \mathbf{f}_c^p$ by Eq. (16.12) locally restore sampled polynomials of degree $p = 2r - 1$. The high-pass p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ locally eliminate such polynomials. Either of the first decomposition level wavelets $\psi^1_{[1]}[k] = h^1[k]$ and $\tilde{\psi}^1_{[1]}[k] = \tilde{h}^1[k]$ has p LDVMs.*

Proof Due to Eq. (16.10), the frequency response of the high-pass p-filter

$$\hat{\tilde{h}}^1[n] = \frac{\omega^{-n}}{\sqrt{2}}\left(1 - \omega^{-n}\,\hat{f}_c^p[n]_1\right) = -\sin^{2r}\frac{\pi n}{N}\,\frac{\omega^{-n}\,\chi^p[n]}{\sqrt{2}}, \quad \hat{\tilde{h}}^1[0] = 0.$$

Then, it follows from Proposition 15.7 that the p-filter $\tilde{\mathbf{h}}^1$ locally eliminates polynomials of degree $p - 1$. The frequency response of the low-pass p-filter

$$\hat{h}^0[n] = \frac{1}{\sqrt{2}}\left(1 + \omega^{-n}\,\hat{f}_c^p[n]_1\right), \quad \hat{h}^0[0] = \sqrt{2} \Longrightarrow \hat{h}^0[n] = \hat{h}^0[0] + \sin^{2r}\frac{\pi n}{N}\,\frac{\chi^p[n]}{\sqrt{2}}.$$

Then, it follows from Proposition 15.5 that the p-filter $\mathbf{h}^0$ locally restores polynomials of degree $p - 1$. The statements concerning the analysis low-pass p-filter $\tilde{\mathbf{h}}^0$ and the synthesis high-pass p-filters $\mathbf{h}^1$ follow directly from Eq. (16.12). ■

Remark 16.2.1 *We emphasize that the low-pass p-filters derived from splines of odd order $p = 2r - 1$ locally restore polynomials of the same degree $p = 2r - 1$ as the*

p-filters derived from splines of even order $p = 2r$, while the corresponding high-pass p-filters eliminate such polynomials. This is a result of the super-convergence property of odd order splines.

16.2.5 Examples of p-filters Derived from Polynomial Splines

In the forthcoming examples we present p-filters derived from polynomial splines of different order using the lifting scheme. Actually, all one needs to know in order to implement the lifting scheme of direct and inverse wavelet transforms, is the frequency response $\hat{f}_c^p[n]_1$ of the prediction p-filter $\mathbf{f_c^p}[\mathbf{n}]_\mathbf{1}$. To derive $\hat{f}_c^p[n]_1$, it is sufficient to calculate the DFT of the B-spline of order p, sampled on the grids $\{k\}$ and $\{k + 1/2\}$.

16.2.5.1 Example: Linear Interpolating Spline, $p = 2$

The frequency responses are

$$\begin{aligned}
&\hat{f}_c^2[n]_1 = \frac{1+\omega^{2n}}{2}, \quad \hat{\tilde{h}}^1[n] = \frac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^n\,\hat{f}_d^{2r}[-n]_1\right) = \sqrt{2}\,\omega^{-n}\,\sin^2\frac{\pi n}{N},\\
&\hat{h}^0[n] = \frac{1+\omega^{-n}\,\hat{f}_c^2[n]_1}{\sqrt{2}} = \sqrt{2}\,\cos^2\frac{\pi n}{N},\\
&\hat{\tilde{h}}^0[n] = \frac{1}{\sqrt{2}}\left(1+\omega^{-n}\,\hat{f}_c^2[n]_1 + W^2[-n]_1\right) = \sqrt{2}\,\cos^2\frac{\pi n}{N}\left(1+2\sin^2\frac{\pi n}{N}\right)\\
&\hat{h}^1[n] = \frac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^n\,\hat{f}_c^2[-n]_1 + W^2[n]_1\right) = \sqrt{2}\,\omega^{-n}\,\sin^2\frac{\pi n}{N}\left(1+2\cos^2\frac{\pi n}{N}\right).
\end{aligned} \tag{16.13}$$

Comments: Comparing Eq. (16.13) with Eqs. (15.34) and (15.36), we observe that, up to normalization, the designed p-filters coincide with the 5/3 p-filters. The low-pass p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore first degree polynomials, while the high-pass p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^1$ locally eliminate first degree polynomials. The wavelets $\psi^1_{[1]}[k] = h^1[k]$ and $\tilde{\psi}^1_{[1]}[k] = \tilde{h}^1[k]$ have two LDVMs each. Impulse and magnitude responses of the p-filters are displayed in Fig. 15.2.

16.2.5.2 Example: Quadratic Interpolating Spline, $p = 3$

The frequency response of the prediction p-filter is

$$\hat{f}_c^3[n]_1 = 4\,\frac{1+w^{2n}}{\omega^{2n}+6+\omega^{-2n}} = \omega^n\,\frac{\cos^4\pi n/N - \sin^4\pi n/N}{\cos^4\pi n/N + \sin^4\pi n/N}, \quad (16.14)$$
$$1-\omega^{-n}\,\hat{f}_c^3[n]_1 = \frac{2\sin^4\pi n/N}{\cos^4\pi n/N + \sin^4\pi n/N}.$$

Therefore, the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore the sampled cubic polynomials while the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^0$ locally eliminate them. Either of the wavelets $\psi^1_{[1]}$ and $\tilde{\psi}^1_{[1]}$ has four LDVMs.

Although quadratic splines consist of pieces of polynomials of second degree, the low-pass p-filters locally restore sampled polynomials of third degree, while the high-pass p-filters locally eliminate them. This fact stems from the super-convergence property of the odd order splines. Figure 16.1 displays the impulse and frequency responses of the p-filters derived from the quadratic interpolating splines.

16.2.5.3 Example: Cubic Interpolating Spline, $p = 4$

The frequency response of the prediction p-filter is

$$\hat{f}_c^4[n]_1 = \omega^n\,\frac{\omega^{3n}+23\omega^n+23\omega^{-n}+\omega^{-3n}}{8(\omega^{2n}+4+\omega^{-2n})}, \quad (16.15)$$
$$1-\omega^{-n}\,\hat{f}_c^4[n]_1 = \frac{\sin^4\pi n/N\,(2+\cos 2\pi n/N)}{2+\cos 4\pi n/N}.$$

Therefore, like in the quadratic spline example, the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore sampled cubic polynomials, while the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^0$ locally eliminate them. Either of the wavelets $\psi^1_{[1]}$ and $\tilde{\psi}^1_{[1]}$ has four LDVMs.

Figure 16.1 displays the impulse and frequency responses of the p-filters derived from the cubic interpolating splines.

16.2.5.4 Example: Interpolating Spline of Fourth Degree, $p = 5$

The frequency response of the prediction p-filter is

$$\hat{f}_c^5[n]_1 = 16\omega^n\,\frac{\omega^{3n}+11\omega^n+11\omega^{-n}+\omega^{-3n}}{\omega^{4n}+76\omega^n+230+76\omega^{-n}+\omega^{-2n}}, \quad (16.16)$$
$$1-\omega^{-n}\,\hat{f}_c^5[n]_1 = \frac{\sin^6\pi n/N\,(10+2\cos 2\pi n/N)}{5+18\cos^2 2\pi n/N+\cos^4 2\pi n/N}.$$

Therefore, the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore sampled fifth degree polynomials while the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^0$ locally eliminate them. Either of the wavelets $\psi^1_{[1]}$ and

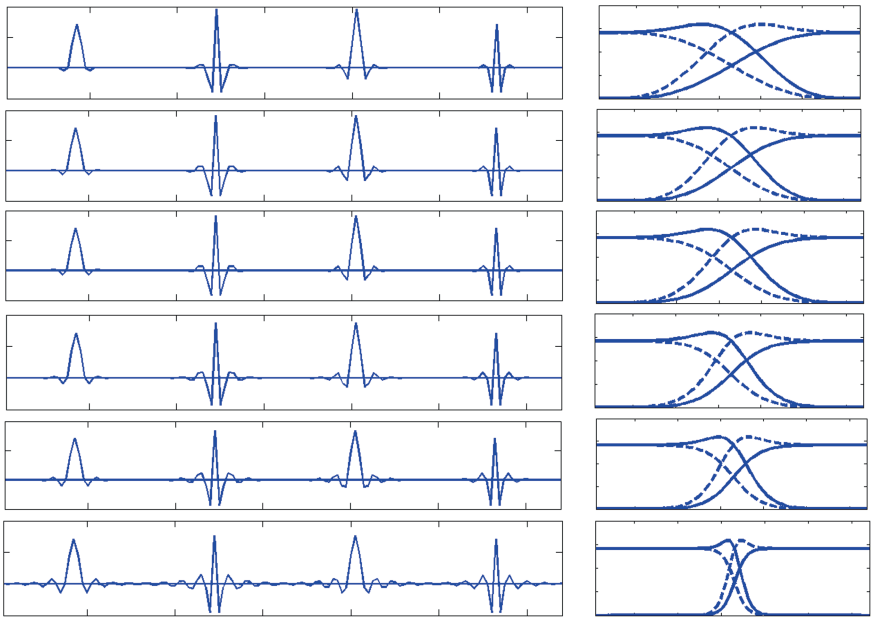


Fig. 16.1 Impulse and magnitude responses of p-filters derived from polynomial splines. Top to bottom: quadratic quasi-interpolating $\searrow$ quadratic upgraded quasi-interpolating $\searrow$ quadratic interpolating $\searrow$ cubic interpolating $\searrow$ interpolating of order 5 $\searrow$ interpolating of order 12. Left panel – impulse response of p-filters left to right: $\mathbf{h}^0 \longrightarrow \mathbf{h}^1 \longrightarrow \tilde{\mathbf{h}}^0 \longrightarrow \tilde{\mathbf{h}}^1$. Right panel – MRs of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ – *dashed lines*, of the p-filters $\tilde{\mathbf{h}}^0$ and $\tilde{\mathbf{h}}^1$ – *solid lines*

$\tilde{\psi}^1_{[1]}$ has six LDVMs. Like in the quadratic spline example, the super-conversion property contributes to the approximation abilities of the p-filter bank.

Figure 16.1 displays the impulse and frequency responses of the p-filters derived from the interpolating splines of fourth degree (fifth order).

All the p-filters originating from interpolating splines except for the linear spline have infinite impulse response (IIR). The high- pass FIR p-filters derived from the linear spline have only two vanishing moments. However, it is possible to design FIR linear phase filters which use local quasi-interpolatory splines and have more vanishing moments. Quasi-interpolating splines are studied in [4, 12]. We outline the design of the prediction p-filter originating from a quadratic quasi-interpolating spline. Design of p-filters that use quasi-interpolating splines of higher order is similar. More details on quasi-interpolating splines will be provided in Volume II of this book.

16.2.5.5 Example: Quadratic Quasi-interpolating Spline

The quadratic spline, which interpolates the even polyphase component $\mathbf{x}_0$ of a signal $\mathbf{x}_0$, is represented as $S^3(t) = \sum_{k=0}^{N/2-1} q[k]\, B^3(t-k)$, where, due to Eq. (16.7), the

DFT of the coefficients is

$$\hat{q}[n]_1 = \frac{\hat{x}_0[n]_1}{u^3[n]_1} = \frac{\hat{x}_0[n]_1}{1 - \frac{1}{2}\,\sin^2 2\pi n/N} = \hat{x}_0[n]_1 \sum_{l=0}^{\infty} \frac{1}{2^l}\,\sin^{2k}\frac{2\pi n}{N}. \tag{16.17}$$

Take two initial terms in the series in Eq. (16.17) and denote,

$$U^3[n]_1 \stackrel{\text{def}}{=} 1 + \frac{1}{2}\,\sin^2\frac{2\pi n}{N} = \frac{-\omega^{2n} + 10 - \omega^{-2n}}{8}. \tag{16.18}$$

The spline $S^3(t) = \sum_{k=0}^{N/2-1} q[k]\,B^3(t-k)$, where the DFT of the coefficients is $\hat{q}[n]_1 = \hat{x}_0[n]_1 U^3[n]_1$, is called the quadratic spline quasi-interpolating the signal $\mathbf{x}_0 \in \Pi[N/2]$. Denote the values of the spline, $s_0[k] \stackrel{\text{def}}{=} S^3(k)$ and $s_1[k] \stackrel{\text{def}}{=} S^3(k+1/2)$, $\kappa \in \mathbb{Z}$. Then, the DFT $\hat{s}_0[n]_1 = g_q^3[n]_1\,\hat{x}_0[n]_1$ and $\hat{s}_1[n]_1 = f_q^3[n]_1\,\hat{x}_0[n]_1$, where

$$g_q^3[n]_1 = u^3[n]_1\,U^3[n]_1 = \frac{-\omega^{-4n} + 4\omega^{-2n} + 58 + 4\omega^{2n} - \omega^{4n}}{64}, \tag{16.19}$$

$$f_q^3[n]_1 = \frac{-\omega^{-2n} + 9 + 9\omega^{2n} - \omega^{4n}}{16}, \tag{16.20}$$

$$\begin{aligned} 1 - \omega^{-n}\,f_q^3[n]_1 &= \frac{\omega^{-3n} - 9\omega^{-n} + 16 - 9\omega^{n} + \omega^{3n}}{16} \\ &= 2\,\sin^4\frac{\pi n}{N}\left(1 + 2\cos^2\frac{\pi n}{N}\right). \end{aligned}$$

Therefore, if the FIR p-filter $\mathbf{f_q^3}$, whose frequency response is given in Eq. (16.20), is employed as a prediction p-filter in the lifting scheme then the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore sampled cubic polynomials while the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^0$ locally eliminate them. Either of the wavelets $\psi_{[1]}^1$ and $\tilde{\psi}_{[1]}^1$ has four LDVMs. This means, in particular, that, like the quadratic interpolating spline, the quadratic quasi-interpolating spline possesses the super-convergence property.

Figure 16.1 displays the impulse and frequency responses of the p-filters derived from the quadratic quasi-interpolatory splines.

Example: upgraded quadratic quasi-interpolating spline

There is a way to enhance the super-convergence property of the quadratic quasi-interpolating spline, which is described in [4, 11]. This is done by upgrading the prediction p-filter $\mathbf{f_q^3}$ to a p-filter $\mathbf{f_u^3}$, whose frequency response is given as

$$\hat{f}_u^3[n]_1 = \hat{f}_q^3[n]_1 + \frac{3}{128}\left(\omega^{2n} - 2 + \omega^{-2n}\right)^2$$
$$= \omega^n \frac{3\omega^{-5n} - 25\omega^{-3n} + 150\omega^{-n} + 150\omega^n - 25\omega^{3n} + 3\omega^{5n}}{256}, \quad (16.21)$$
$$1 - \omega^{-n} f_u^3[n]_1 = \sin^6 \frac{\pi n}{N}\left(8 + 9\cos\frac{2\pi n}{N} + 3\cos^2\frac{2\pi n}{N}\right).$$

Therefore, if the FIR p-filter $\mathbf{f_u^3}$, whose frequency response is given in Eq. (16.21), is taken as the prediction p-filter in the lifting scheme then the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$ locally restore the sampled polynomials of fifth degree while the p-filters $\mathbf{h}^1$ and $\tilde{\mathbf{h}}^0$ locally eliminate them. Either of the wavelets $\psi_{[1]}^1$ and $\tilde{\psi}_{[1]}^1$ has six LDVMs.

Figure 16.1, which was produced by the MATLAB code `BW_exam_5P7.m`, displays impulse and magnitude responses (MR) of analysis and synthesis low- and high- pass p-filters derived from polynomial splines of different orders.

One can observe that all the impulse responses are symmetric and well localized in time domain, although most of them are not finite (up to periodization). The MRsw are flat, especially for the p-filters derived from the twelfth order spline, whose shapes are near rectangular. The MRs of the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^1$ mirror each other. The same is true for the pair $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^1$. Note the "bumps" which have the MRs of the p-filters $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^1$ near the cutoff frequency.

16.3 Prediction Filters Derived from Discrete Splines

The prediction filters are derived from discrete splines by a scheme similar to the derivation scheme for the polynomial splines. To be specific, discrete splines of span 2, which interpolate even samples of a signal are constructed. Then, odd samples are predicted by the midpoint values of the splines.

16.3.1 Summary for the Discrete Splines of Span 2

Discrete splines are described in details in Chap. 13. Here we gather facts related to the special case of discrete splines $\mathbf{s}_{[1]}^{2r} \in {}^{2r}\mathscr{S}_{[1]}$ with the span 2 ($m = 1$).

The N-periodic discrete B-spline $\mathbf{b}^2$ of second order is produced by$N$-periodization of the sequence $\beta[0] = 1/2,\ \beta[\pm 1] = 1/4$. The DFT is $\hat{b}^2[k] = \cos^2 \pi n/N$. The discrete B-splines of higher even orders are defined iteratively via the discrete circular convolution:

$$\mathbf{b}^{2r} \stackrel{\text{def}}{=} \mathbf{b}^2 \circledast \mathbf{b}^{2(r-1)} \Longrightarrow \hat{\mathbf{b}}^{2r}[n] = \cos^{2r}\frac{\pi n}{N}.$$

The N-periodic discrete splines $\mathbf{s}^{2r}$ are defined as linear combinations of the B-splines:

$$s^{2r}[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} q[l] b^{2r}[k-2l] \Longrightarrow \hat{\mathbf{s}}^{2r}[n] = \hat{q}[n]_1 \cos^{2r} \frac{\pi n}{N}.$$

Since the sequence $\hat{q}[n]_1$ is $N/2$-periodic, the DFT of the even $\mathbf{s}_0^{2r}$ and the odd $\mathbf{s}_1^{2r}$ polyphase components of the discrete spline $\mathbf{s}^{2r}$ are

$$\begin{aligned}
\hat{s}_0^{2r}[n]_1 &= \frac{\hat{s}^{2r}[n] + \hat{s}^{2r}[n+N/2]}{2} = \frac{\hat{q}[n]_1}{2} U^{2r}[n]_1, \\
\hat{s}_1^{2r}[n]_1 &= \frac{\hat{s}^{2r}[n] - \hat{s}^{2r}[n+N/2]}{2\omega^{-n}} = \frac{\hat{q}[n]_1}{2} V^{2r}[n]_1, \\
U^{2r}[n]_1 &\stackrel{\text{def}}{=} \cos^{2r} \frac{\pi n}{N} + \sin^{2r} \frac{\pi n}{N}, \\
V^{2r}[n]_1 &\stackrel{\text{def}}{=} \omega^n \left(\cos^{2r} \frac{\pi n}{N} - \sin^{2r} \frac{\pi n}{N} \right).
\end{aligned} \tag{16.22}$$

16.3.2 Filter Banks

Mimicking the p-filter design scheme based on polynomial splines, we assume that a discrete spline $\mathbf{s}^{2r}$ interpolates the even polyphase component $\mathbf{x}_0$ of a signal $\mathbf{x} \in \mathit{\Pi}[N]$ on the grid $\{2k\}$ and predict the component $\mathbf{x}_1$ via the spline values on the grid $\{2k+1\}$. Thus,

$$\mathbf{s}_0^{2r} = \mathbf{x}_0 \Longrightarrow \hat{q}[n]_1 = \frac{2\,\hat{x}_0[n]_1}{U^{2r}[n]_1}, \quad \hat{s}_1^{2r}[n]_1 = \frac{V^{2r}[n]_1}{U^{2r}[n]_1}\,\hat{x}_0[n]_1.$$

The sequence $U^{2r}[n]_1$ is strictly positive, symmetric about $K/2 = N/4$ and $U^{2r}[0] = 1$. Similarly to Eq. (16.8), we define the prediction p-filters $\mathbf{f}_d^{2r}$ via the frequency responses:

$$\hat{f}_d^{2r}[n]_1 \stackrel{\text{def}}{=} \frac{V^{2r}[n]_1}{U^{2r}[n]_1} = \frac{\omega^n \left(\cos^{2r} \pi n/N - \sin^{2r} \pi n/N \right)}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \tag{16.23}$$

Obviously, $\omega^{-n}\, \hat{f}_d^{2r}[n]_1 = \omega^n\, \hat{f}_d^{2r}[-n]_1$ and

$$1 - \omega^{-n} \hat{f}_d^{2r}[n]_1 = \frac{\sin^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \tag{16.24}$$

If the p-filter $\mathbf{f}_d^{2r}$, whose frequency response is given in Eq. (16.23), is used as a prediction filter in the lifting scheme then the PR p-filter bank is

$$\begin{aligned}
&\hat{h}^0[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}\left(1+\omega^{-n}\hat{f}_d^{2r}[n]_1\right) = \frac{\sqrt{2}\cos^{2r}\pi n/N}{\cos^{2r}\pi n/N+\sin^{2r}\pi n/N},\\
&\hat{\tilde{h}}^1[n] \stackrel{\text{def}}{=} \frac{\omega^{-n}}{\sqrt{2}}\left(1-\omega^{n}\hat{f}_d^{2r}[-n]_1\right) = \frac{\omega^{-n}\sqrt{2}\sin^{2r}\pi n/N}{\cos^{2r}\pi n/N+\sin^{2r}\pi n/N} \qquad (16.25)\\
&\hat{\tilde{h}}^0[n] = \hat{h}^0[n]\left(1+\sqrt{2}\,\omega^n\hat{\tilde{h}}^1[n]\right), \quad \hat{h}^1[n] = \hat{\tilde{h}}^1[n]\left(1+\sqrt{2}\,\omega^n\hat{h}^0[n]\right).
\end{aligned}$$

All the p-filters have a linear phase. Except for the simplest case $r = 1$, the impulse responses of the p-filters are infinite. Nevertheless, they are well localized in time domain. The following Proposition, whose proof is similar to the proof of Proposition 16.4, is the consequence of Eq. (16.24).

Proposition 16.5 *Assume that the discrete spline order is* $2r$. *The low-pass p-filters* $\mathbf{h}^0$ *and* $\tilde{\mathbf{h}}^0$ *derived from the prediction p-filter* $\tilde{\mathbf{f}} = \mathbf{f}_d^p$ *by Eq.(16.25) locally restore sampled polynomials of degree* $p = 2r-1$. *The high-pass p-filters* $\mathbf{h}^1$ *and* $\tilde{\mathbf{h}}^1$ *locally eliminate such polynomials. Consequently, either of the wavelets* $\psi^1_{[1]}[k] = h^1[k]$ *and* $\tilde{\psi}^1_{[1]}[k] = \tilde{h}^1[k]$ *have* $2r$ *LDVMs.*

We emphasize that Eq. (16.25) provide an explicit expression for the p-filters with arbitrary approximation accuracy.

Remark 16.3.1 *The sequences* $\hat{h}[n]$ *and* $\omega^n\,\hat{\tilde{h}}^1[n]$ *are the frequency responses of the periodic half-band low- and high-pass Butterworth filters, respectively. The Butterworth filters are widely used in signal processing ([6]).*

Due to the relation of the p-filters to the Butterworth p-filters, it is natural to refer to the wavelets derived from discrete splines as the periodic *Butterworth wavelets*. .

Remark 16.3.2 *One-pass non-periodic Butterworth filters were used in [5] for the design of orthogonal non-symmetric wavelets . Computations in [5] were conducted in the time domain using recursive filtering. Biorthogonal periodic wavelets derived from discrete splines are presented in [1], while non-periodic biorthogonal wavelets are introduced in [2].*

16.3.2.1 Examples

Spline of second order: $r = 1$

$$\hat{f}_d^2[n]_1 = \omega^n\left(\cos^2\pi n/N - \sin^2\pi n/N\right) = \frac{1+\omega^{2n}}{2} = \hat{f}_c^2[n]_1.$$

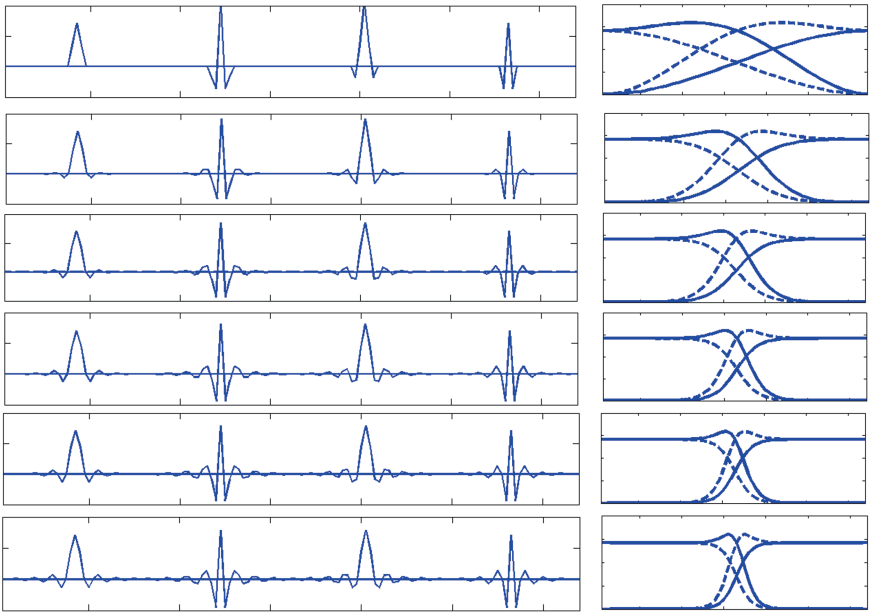


Fig. 16.2 Impulse and magnitude responses of p-filters derived from discrete splines of orders $2r$, $r = 1$ (*top*) to $r = 6$ (*bottom*). Left panel – impulse response of p-filters. Left to right: $\mathbf{h}^0 \longrightarrow \mathbf{h}^1 \longrightarrow \tilde{\mathbf{h}}^0 \longrightarrow \tilde{\mathbf{h}}^1$. Right panel – MRs of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ – *dashed lines*, of the p-filters $\tilde{\mathbf{h}}^0$ and $\tilde{\mathbf{h}}^1$ – *solid lines*

Thus, the prediction p-filter derived from the discrete spline of second order coincides with the p-filter derived from the second order polynomial spline.

Splines of fourth order: $r = 2$

$$\hat{f}_d^4[n]_1 = \frac{\omega^n \left(\cos^4 \pi n/N - \sin^4 \pi n/N\right)}{\cos^4 \pi n/N + \sin^4 \pi n/N} = \hat{f}_c^3[n]_1. \tag{16.26}$$

Comparing Eq. (16.26) with Eq. (16.14), we observe that the prediction p-filter derived from the discrete spline of fourth order coincides with the p-filter derived from the quadratic polynomial spline. Consequently, all the analysis and synthesis p-filters are the same for these two kinds of splines. Such a coincidence does not take place for higher order splines. Figure 16.2, which was produced by the MATLAB code `BW_exam_5P7.m`, displays impulse and magnitude responses (MR) of analysis and synthesis low- and high- pass p-filters derived from discrete splines of different orders.

Implementation of transforms

Forward and inverse biorthogonal wavelet transforms of periodic signals with the designed filter banks are implemented by the MATLAB functions `BW_anal.m` and `BW_synth.m`, respectively. Two-dimensional transforms are implemented by subsequent application of the 1D transforms to columns and rows of the 2D arrays. This is done by the MATLAB functions `BW_2D_anal.m` and `BW_2D_synth.m`.

References

1. A. Averbuch, A.B. Pevnyi, V. Zheludev, Biorthogonal butterworth wavelets derived from discrete interpolatory splines. IEEE Trans. Signal Process. **49**(11), 2682–2692 (2001)
2. A. Averbuch, A.B. Pevnyi, V. Zheludev, Butterworth wavelet transforms derived from discrete interpolatory splines: recursive implementation. Signal Process. **81**(11), 2363–2382 (2001)
3. A. Averbuch, V. Zheludev, Construction of biorthogonal discrete wavelet transforms using interpolatory splines. Appl. Comput. Harmon. Anal. **12**(1), 25–56 (2002)
4. A. Averbuch, V. Zheludev, Wavelet transforms generated by splines. Int. J. Wavelets Multiresolut. Inf. Process. **5**(2), 257–291 (2007)
5. C. Herley, M. Vetterli, Wavelets and recursive filter banks. IEEE Trans. Signal Process. **41**(12), 2536–2556 (1993)
6. A.V. Oppenheim, R.W. Schafer, *Discrete-time signal processing*, 3rd edn. (Prentice Hall, New York, 2010)
7. I.J. Schoenberg, Contributions to the problem of approximation of equidistant data by analytic functions. Q. Appl. Math. **4**, 45–99, 112–141, (1946), parts A and B.
8. W. Sweldens, in *Lifting scheme: a new philosophy in biorthogonal wavelet constructions*, eds. by A.F. Laine, M.A. Unser. Wavelet Applications in Signal and Image Processing III, vol. 2569 of Proc. SPIE (1995), pp. 68–79
9. W. Sweldens, The lifting scheme: a custom-design construction of biorthogonal wavelets. Appl. Comput. Harmon. Anal. **3**(2), 186–200 (1996)
10. W. Sweldens, The lifting scheme: a construction of second generation wavelets. SIAM J. Math. Anal. **29**(2), 511–546 (1997)
11. V. Zheludev, Local quasi-interpolatory splines and fourier transforms. Sov. Math. Dokl. **31**, 573–577 (1985)
12. V. Zheludev, Local spline approximation on a uniform mesh. U.S.S.R. Comput. Math. Math. Phys. **27**(5), 8–19 (1987)
13. V. Zheludev, Periodic splines and the fast fourier transform. Comput. Math. Math. Phys. **32**(2), 149–165 (1992)

Chapter 17
Wavelet Frames Generated by Spline Based p-Filter Banks

Abstract This chapter presents a design scheme to generate tight and so-called semi-tight frames in the space of discrete-time periodic signals. The frames originate from three- and four-channel perfect reconstruction periodic filter banks. The filter banks are derived from interpolating and quasi-interpolating polynomial splines and from discrete splines. Each filter bank comprises one linear phase low-pass filter (in most cases interpolating) and one high-pass filter, whose magnitude's response mirrors that of a low-pass filter. In addition, these filter banks comprise one or two band-pass filters. In the semi-tight frames case, all the filters have linear phase and (anti)symmetric impulse response, while in the tight frame case, some of band-pass filters are slightly asymmetric. The design scheme enables to design framelets with any number of LDVMs.

The computational complexity of the framelet transforms practically does not depend on the number of LDVMs and on the size of the impulse response of filters. Recently frames or redundant expansions of signals have attracted considerable interest from researchers working in signal processing although one particular class of frames, the Gabor systems, has been applied and investigated since 1946 [11]. As the requirement of one-to-one correspondence between the signal and its transform coefficients is dropped, there is more freedom to design and implement frame transforms. Frame expansions of signals demonstrate resilience to quantization noise and to coefficients losses [12–14]. Thus, frames may serve as a tool for error correction of signals that are transmitted through lossy/noisy channels. Recently, overcomplete representation of signals was applied to image reconstruction [5, 6]. Combination of wavelet frames with the Bregman iterations techniques [2, 19] provided a new impact to the image processing applications such as deconvolution, inpainting, denoising, to name a few [3, 4, 9, 17, 18]. It was mentioned in Sect. 2.2 that oversampled PR filter banks generate a specific kind of frames in the periodic signal space. This chapter presents families of frames whose generating three- and four-channel p-filter banks originate from polynomial and discrete splines. These frames have properties such as symmetry, interpolation, and flat spectra, which are attractive for signal processing. These

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_17,

properties are combined with fine time-domain localization and efficient implementation. The families include framelets that have any number of discrete vanishing moments. Non-compactness of their support is compensated by exponential decay as time tends to infinity.

In some applications, it is important to have compactly supported framelets, which are provided by filter banks with FIR filters. However, it was proved in [15] that a 3-channel filter bank with linear phase interpolatory filters can only produce a tight frame whose high-frequency framelets have two and one vanishing moments. However, it is possible to construct a variety of compactly supported interpolatory symmetric, referred to as *semi-tight* frames, with increased number of vanishing moments. Adding one more channel to the filter bank makes it possible to construct compactly supported interpolating symmetric tight frames [7].

17.1 Oversampled PR Filter Banks and Frames

In this section, we discuss 3- and 4-channel perfect reconstruction (PR) p-filter banks with downsampling factor of 2. These p-filter banks generate frames in the space $\Pi[N]$ of N-periodic signals.

17.1.1 Oversampled p-Filter Banks with Downsampling Factor of 2

A set of p-filters $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s\right\}$, $s = 0, \ldots, S-1$, which, being time-reversed and applied to an input signal $\mathbf{x} \in \Pi[N]$, produces the set of the output signals $\{\mathbf{y}^s\} \in \Pi[N/2]$, $s = 0, \ldots, S-1$, which are downsampled by factor of 2,

$$y^s[l] = \sum_{k=0}^{N-1} \tilde{h}^s[k-2l]\, x[k], \quad s = 0, \ldots, S-1, \ l \in \mathbb{Z}, \tag{17.1}$$

is called S-channel analysis p-filter bank. A set of p-filters $\mathbf{H} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s\right\}$, $s = 0, \ldots, S-1$, which, being applied to a set of input signals $\{\mathbf{y}^s\}$ from $\Pi[N/2]$, $s = 0, \ldots, S-1$, that are upsampled by factor of 2, produces the output signal

$$\check{x}[l]) = \sum_{s=0}^{S-1} \sum_{k=0}^{N/2-1} h^s[l-2k]\, y^s[k] \ \ l \in \mathbb{Z}, \tag{17.2}$$

belonging to $\Pi[N]$, is called S-channel synthesis p-filter bank. If the upsampled signals $\{\mathbf{y}^s\} \in \Pi[N/2]$, $s = 0, \ldots, S-1$, which are defined in Eq. (17.1), are used as an input to the synthesis p-filter bank and the output signal is $\check{\mathbf{x}} = \mathbf{x}$, then the pair of analysis–synthesis p-filter banks form a perfect reconstruction (PR) p-filter bank.

If the number of channels S equals to the downsampling factor of 2 then the p-filter bank is said to be critically sampled. If $S > 2$ then the p-filter bank is oversampled. Critically sampled PR p-filter banks generate biorthogonal wavelet bases (Chap. 15), while oversampled PR p-filter banks serve as a source for the design of discrete-time wavelet frames.

As in Chap. 15, $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^0$ are assumed to be low-pass p-filters.

Let's assume that $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s\right\}$, $s = 0, \ldots, S-1$, is an analysis p-filter bank with the downsampling factor of 2. Then, its application to a signal $\mathbf{x} \in \Pi[N]$ produces S signals $\{\mathbf{y}^s\} \in \Pi[N/2]$, $s = 0, \ldots, S-1$, which are defined in Eq. (17.1). It follows from Eq. (2.1) that

$$\hat{y}^s[n]_1 = \hat{\tilde{h}}_0^s[-n]_1\, \hat{x}_0[n]_1 + \hat{\tilde{h}}_1^s[-n]_1\, \hat{x}_1[n]_1, \tag{17.3}$$

where $\hat{x}_p[n]_1$ and $\hat{\tilde{h}}_p^s[n]_1$, $p = 0, 1,\ s = 0, \ldots, S-1$, are the DFT of the polyphase components of the signal $\check{\mathbf{x}}$ and the p-filters $\left\{\tilde{\mathbf{h}}^s\right\}$, respectively.

Assume now that $\mathbf{H} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s\right\}$, $s = 0, \ldots, S-1$, is a synthesis p-filter bank with the upsampling factor of 2. Then, its application to the upsampled signals $\mathbf{y}^s$ produces a signal $\check{\mathbf{x}}$ from $\Pi[N]$ defined in Eq. (17.2).

Equation (2.2) implies that the DFT of the polyphase components of the signal $\check{\mathbf{x}}$ are

$$\hat{\check{x}}_0[n]_1 = \sum_{s=0}^{S-1} \hat{h}_0^s[n]_1\, \hat{y}^s[n]_1, \quad \hat{\check{x}}_1[n]_1 = \sum_{s=0}^{S-1} \hat{h}_1^s[n]_1\, \hat{y}^s[n]_1. \tag{17.4}$$

Equations (17.3) and (17.4) can be represented in a matrix form by

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix} = \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{\check{x}}_0[n]_1 \\ \hat{\check{x}}_1[n]_1 \end{pmatrix}, \quad \begin{pmatrix} \hat{\check{x}}_0[n]_1 \\ \hat{\check{x}}_1[n]_1 \end{pmatrix} = \mathbf{P}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix}$$

where the $S \times 2$ analysis and the $2 \times S$ synthesis polyphase matrices are, respectively,

$$\tilde{\mathbf{P}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^0[n]_1 & \hat{\tilde{h}}_1^0[n]_1 \\ \vdots & \vdots \\ \hat{\tilde{h}}_0^{S-1}[n]_1 & \hat{\tilde{h}}_1^{S-1}[n]_1 \end{pmatrix} \quad \mathbf{P}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 \cdots \hat{h}_0^{S-1}[n]_1 \\ \hat{h}_1^0[n]_1 \cdots \hat{h}_1^{S-1}[n]_1 \end{pmatrix}, \quad n \in \mathbb{Z}.$$

If the relations

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \mathbf{I}_2, \tag{17.5}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix, holds for all $n \in \mathbb{Z}$ then

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix} = \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}.$$

Thus, Eq. (17.5) is the condition for the analysis–synthesis pair $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$ of p-filter banks to form a PR p-filter bank.

17.1.2 Frames in the Space of Periodic Signals

Definition 17.1 *A system* $\tilde{\Phi} \stackrel{\text{def}}{=} \{\tilde{\phi}_l\}_{l=0}^{L-1},\ L \geq N$, *of signals from* $\Pi[N]$ *forms a frame of the space* $\Pi[N]$ *if there exist positive constants A and B such that for any signal* $\mathbf{x} = \{x[k]\} \in \Pi[N]$

$$A\|\mathbf{x}\|^2 \leq \sum_{l=0}^{L-1} |\langle \mathbf{x}, \tilde{\phi}_l \rangle|^2 \leq B\|\mathbf{x}\|^2.$$

If the frame bounds A and B are equal to each other then the frame is said to be tight.

If the system $\tilde{\Phi}$ is a frame then there exists another frame $\Phi \stackrel{\text{def}}{=} \{\phi_l\}_{l=0}^{L-1}$ of the space $\Pi[N]$ such that any signal $\mathbf{x} \in \Pi[N]$ can be expanded into the sum $\mathbf{x} = \sum_{l=0}^{L-1} \langle \mathbf{x}, \tilde{\phi}_l \rangle\, \phi_l$. The analysis $\tilde{\Phi}$ and synthesis Φ frames can be interchanged. Together they form a so-called bi-frame. If the frame is tight then Φ can be chosen as $\Phi = c\tilde{\Phi}$.

If the elements $\{\tilde{\phi}_l\}$ of the analysis frame $\tilde{\Phi}$ are not linearly independent ($L > N$) then many synthesis frames can be associated with a given analysis frame. In this case, the expansion $\mathbf{x} = \sum_{l=0}^{L-1} \langle \mathbf{x}, \tilde{\phi}_l \rangle\, \phi_l$ provides a redundant representation of the signal $\mathbf{x}$.

It was established in [8] that the PR filter banks operating in the space l_1 of decaying discrete-time signals generate frames of this space. A similar fact was proved in [20] for the p-filter banks operating in $\Pi[N]$.

17.1.2.1 One-Level Frame Transform

Assume that an analysis $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s\right\}$ and a synthesis $\mathbf{H} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s\right\}$, $s = 0, \ldots, S-1$ p-filter banks with downsampling by factor of 2, constitute a PR p-filter bank.

We denote,

$$\tilde{\psi}^s_{[1]} \stackrel{\text{def}}{=} \left\{\tilde{\psi}^s_{[1]}[l] = \tilde{h}^s[l]\right\}_{l=0}^{N-1}, \quad \psi^s_{[1]} \stackrel{\text{def}}{=} \left\{\psi^s_{[1]}[l] = h^s[l]\right\}_{l=0}^{N-1}, s = 0, \ldots, S-1, \tag{17.6}$$

where $\left\{\tilde{h}^s[l]\right\}$ and $\{h^s[l]\}$ are the impulse responses of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, respectively. Then, relations in Eqs. (17.1) and (17.2) can be rewritten as

$$x[k] = \sum_{s=0}^{S-1} x^s_{[1]}[k], \quad x^s_{[1]}[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} y^s_{[1]}[l]\, \psi^s_{[1]}[k-2l], \tag{17.7}$$

$$y^s_{[1]}[l] = \sum_{\kappa=0}^{N-1} \tilde{\psi}^s_{[1]}[\kappa - 2l]\, x[\kappa] = \langle \mathbf{x}, \tilde{\psi}^s_{[1]}[\cdot - 2l]\rangle,$$

$$\Longleftrightarrow \mathbf{x} = \sum_{s=0}^{S-1} \mathbf{x}^s_{[1]} = \sum_{s=0}^{S-1} \sum_{l=0}^{N/2-1} \langle \mathbf{x}, \tilde{\psi}^s_{[1]}[\cdot - 2l]\rangle\, \psi^s_{[1]}[\cdot - 2l]. \tag{17.8}$$

Thus, the signal $\mathbf{x} \in \Pi[N]$ becomes expanded over the system $\left\{\psi^s_{[1]}[\cdot - 2l]\right\}$, $s = 0, \ldots, S-1$, $l = 0, \ldots, N/2-1$. Expansion coefficients are the inner products $\left\{\langle \mathbf{x}, \tilde{\psi}^s_{[1]}[\cdot - 2l]\rangle\right\}$.

Theorem 17.1 *[20] If the polyphase matrices* $\tilde{\mathbf{P}}$ *and* $\mathbf{P}$ *of p-filter banks* $\tilde{\mathbf{H}}$ *and* $\mathbf{H}$ *satisfy the PR condition of Eq. (17.5) then the 2-sample shifts* $\left\{\tilde{\psi}^s_{[1]}[\cdot - 2l]\right\}$ *and* $\left\{\psi^s_{[1]}[\cdot - 2l]\right\}$, $s = 0, \ldots, S-1$, $l = 0, \ldots, N/2-1$ *of the signals* $\tilde{\psi}^s_{[1]}$ *and* $\psi^s_{[1]}$, $s = 0, \ldots, S-1$, *defined in Eq. (17.6), form a pair of the analysis and the synthesis frames of the space* $\Pi[N]$, *respectively. These frames are interchangeable. If the relations*

$$\tilde{\mathbf{P}}[n]^T \cdot \tilde{\mathbf{P}}[-n] = c\mathbf{I}_2, \quad n \in \mathbb{Z}, \tag{17.9}$$

hold then the frame is tight.

The notation $\cdot^T$ means matrix transposition. If the condition in Eq. (17.9) is satisfied, then the synthesis filter bank can be chosen to be equal to the analysis one (up to a constant factor).

If the number of channels in p-filter banks $\tilde{\mathbf{H}}$ and $\mathbf{H}$ is $S = 2$, then the number of elements in the frames is N, which equals the dimension of the space $\Pi[N]$. Thus, in that case, the sets $\left\{\tilde{\psi}^s_{[1]}[\cdot - 2l]\right\}$ and $\left\{\psi^s_{[1]}[\cdot - 2l]\right\}$, $s = 0, 1$, $l = 0, \ldots, N/2-1$, constitute a biorthogonal pair of bases of the space $\Pi[N]$. This case was discussed in Chap. 15. If $S > 2$ then the representation of signals from $\Pi[N]$, given in Eq. (17.8), is redundant. The redundancy ratio for one-level frame transform is $\rho = S/2 : 1$.

The signals $\tilde{\psi}^s_{[1]}$ and $\psi^s_{[1]}$, $s = 0, \ldots, S-1$, are called the analysis and synthesis discrete-time *framelets* of the first decomposition level, respectively.

17.1.2.2 Multi-Level Frame Transform

To increase redundancy of a signal representation, the frame transform is applied to the low frequency signal

$$\mathbf{y}^0_{[1]} = \left\{ y^0_{[1]}[l] \right\}, \quad y^0_{[1]}[l] = \sum_{k=0}^{N-1} \tilde{\psi}^s_{[1]}[k-2l]\, x[k],$$

which belongs to $\Pi[N/2]$.

Assume that a pair $\tilde{\mathbf{H}}_{[1]} \stackrel{\text{def}}{=} \left\{ \tilde{\mathbf{h}}^s_{[1]} \right\}$ and $\mathbf{H}_{[1]} \stackrel{\text{def}}{=} \left\{ \mathbf{h}^s_{[1]} \right\}$, $s = 0, \ldots, S-1$, is a PR p-filter bank, which operates in the subspace $\Pi[N/2]$. It means that the IRs of the filters $\left\{ \tilde{h}^s_{[1]}[k] \right\}$ and $\left\{ h^s_{[1]}[k] \right\}$, $k \in \mathbb{Z}$, are $N/2$-periodic signals.

Remark 17.1 *Suppose,* $\tilde{\mathbf{P}}[n]$ *and* $\mathbf{P}[n]$, $n \in \mathbb{Z}$, *are the polyphase matrices of the p-filter banks* $\tilde{\mathbf{H}}$ *and* $\mathbf{H}$, *respectively. It is natural to define the p-filter banks* $\tilde{\mathbf{H}}_{[1]}$ *and* $\mathbf{H}_{[1]}$ *via their polyphase matrices as* $\tilde{\mathbf{P}}_{[1]}[n] \stackrel{\text{def}}{=} \tilde{\mathbf{P}}[2n]$, $\mathbf{P}_{[1]}[n] \stackrel{\text{def}}{=} \mathbf{P}[2n]$. *Surely, the PR condition* $\mathbf{P}_{[1]}[n] \cdot \tilde{\mathbf{P}}_{[1]}[-n] = \mathbf{I}_2$ *is satisfied. However, different p-filter banks are possible.*

Similarly to the whole signal $\mathbf{x}$, the low-frequency signal $\mathbf{y}^0_{[1]}$ can be represented as follows:

$$\begin{aligned}
y^0_{[1]}[k] &= \sum_{s=0}^{S-1} \sum_{l=0}^{N/4-1} y^s_{[2]}[l]\, h^s_{[1]}[k-2l], \\
y^s_{[2]}[l] &= \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu-2l]\, y^0_{[1]}[\nu] = \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu-2l] \sum_{\kappa=0}^{N-1} \tilde{\psi}^0_{[1]}[\kappa-2\nu]\, x[\kappa] \\
&= \sum_{\kappa=0}^{N-1} x[\kappa] \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu-2l]\, \tilde{\psi}^0_{[1]}[\kappa-2\nu] \\
&= \sum_{\kappa=0}^{N-1} x[\kappa] \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu]\, \tilde{\psi}^0_{[1]}[\kappa-2\nu-4l] \\
&= \left\langle \mathbf{x}, \tilde{\psi}^s_{[2]}[\cdot - 4l] \right\rangle, \quad \text{where } \tilde{\psi}^s_{[2]}[\kappa] \stackrel{\text{def}}{=} \sum_{\nu=0}^{N/2-1} \tilde{h}^s_{[1]}[\nu]\, \tilde{\psi}^0_{[1]}[\kappa-2\nu].
\end{aligned} \tag{17.10}$$

By substituting Eq. (17.10) into Eq. (17.7) with $s = 0$, we get

$$\begin{aligned}
x^0_{[1]}[k] &= \sum_{l=0}^{N/2-1} y^0_{[1]}[l]\, \psi^0_{[1]}[k-2l] \\
&= \sum_{l=0}^{N/2-1} \psi^0_{[1]}[k-2l] \sum_{s=0}^{S-1} \sum_{\lambda=0}^{N/4-1} y^s_{[2]}[\lambda]\, h^s_{[1]}[l-2\lambda] \\
&= \sum_{s=0}^{S-1} \sum_{\lambda=0}^{N/4-1} y^s_{[2]}[\lambda] \sum_{l=0}^{N/2-1} \psi^0_{[1]}[k-2l]\, h^s_{[1]}[l-2\lambda] \\
&= \sum_{s=0}^{S-1} \sum_{\lambda=0}^{N/4-1} y^s_{[2]}[\lambda]\ \psi^s_{[2]}[k-4\lambda], \\
&\text{where } \psi^s_{[2]}[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} \psi^0_{[1]}[k-2l]\, h^s_{[1]}[l], \quad s = 0, \ldots, S-1.
\end{aligned}$$

The signals $\left\{\tilde{\psi}^s_{[2]}\right\}$ and $\left\{\psi^s_{[2]}\right\}$ belonging to $\Pi[N/4]$, are called the analysis and the synthesis discrete time periodic framelets of the second decomposition level, respectively.

Resulting from the two-level framelet transform, we have the signal $\mathbf{x}$ expanded over 4-sample shifts of the second level framelets and 2-sample shifts of the first level framelets (except for the low-frequency framelet): $\mathbf{x} = \sum_{s=0}^{S-1} \mathbf{x}^s_{[2]} + \sum_{s=1}^{S-1} \mathbf{x}^s_{[1]}$, where

$$\begin{aligned}
x^s_{[2]}[k] &= \sum_{l=0}^{N/4-1} \left\langle \mathbf{x},\, \tilde{\psi}^s_{[2]}[\cdot - 4l] \right\rangle\, \psi^s_{[2]}[k-4l],\ s = 0, \ldots, S-1, \\
x^s_{[1]}[k] &= \sum_{l=0}^{N/2-1} \left\langle \mathbf{x},\, \tilde{\psi}^s_{[1]}[\cdot - 2l] \right\rangle\, \psi^s_{[1]}[k-2l],\ s = 1, \ldots, S-1.
\end{aligned}$$

Assume that a pair $\tilde{\mathbf{H}}_{[\mu]} \stackrel{\text{def}}{=} \left\{\tilde{\mathbf{h}}^s_{[\mu]}\right\}$, $s = 0, \ldots, S-1$, and $\mathbf{H}_{[\mu]} \stackrel{\text{def}}{=} \left\{\mathbf{h}^s_{[\mu]}\right\}$, $s = 0, \ldots, S-1$, form PR p-filter banks, which operate in the subspaces $\Pi[N/2^\mu]$. Let $\tilde{\mathbf{P}}_{[\mu]}[n]$, $\mu = 0, \ldots, m-1$, and $\mathbf{P}_{[\mu]}[n]$ are their polyphase matrices, respectively.

Then, the iterated transform with the polyphase matrices $\tilde{\mathbf{P}}_{[\mu]}[n]$ and $\mathbf{P}_{[\mu]}[n]$ leads to the frame expansion of the signal $\mathbf{x} = \mathbf{x}^0_{[m]} + \sum_{\mu=1}^{m} \sum_{s=1}^{S-1} \mathbf{x}^s_{[\mu]}$, where

$$\begin{aligned}
&x^s_{[m]}[k] = \sum_{l\in\mathbb{Z}} \left\langle \mathbf{x},\, \tilde{\psi}^s_{[m]}[\cdot - 2^m l] \right\rangle\, \psi^s_{[m]}[k-2^m l],\ s = 0, \ldots, S-1, \\
&x^s_{[\mu]}[k] = \sum_{l=0}^{N/2^\mu-1} \left\langle \mathbf{x},\, \tilde{\psi}^s_{[\mu]}[\cdot - 2^\mu l] \right\rangle\, \psi^s_{[\mu]}[k-2^\mu l], \\
&\mu < m,\ s = 1, \ldots, S-1.
\end{aligned} \tag{17.11}$$

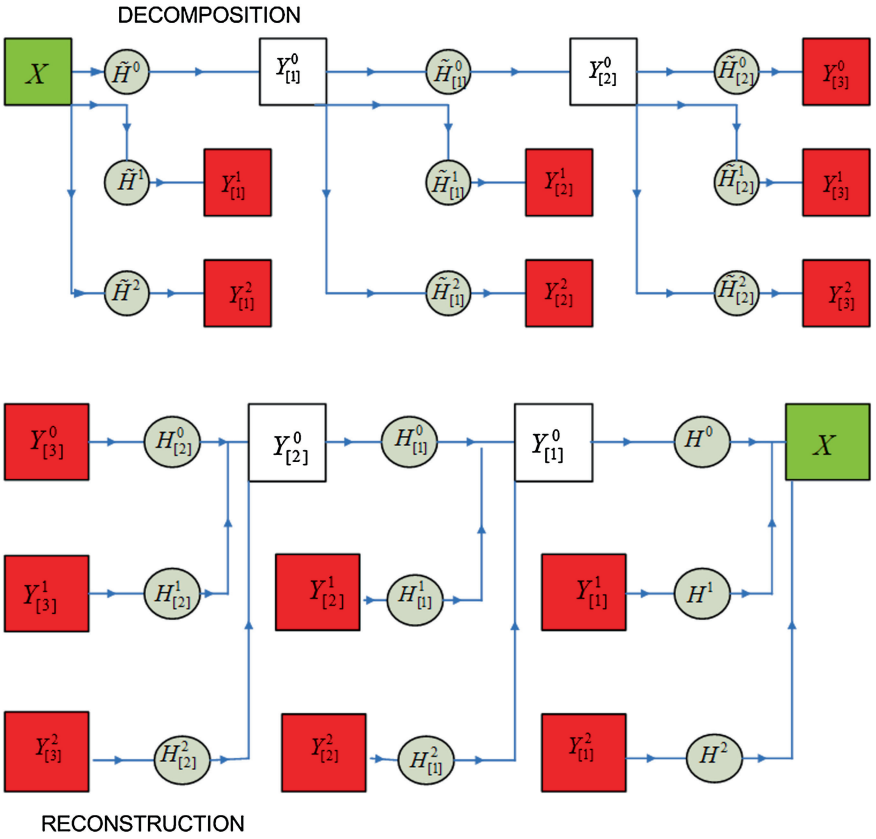


Fig. 17.1 Diagrams of direct (*top*) and inverse (*bottom*) 3-level framelet transforms

The synthesis and analysis framelets are derived iteratively

$$\begin{aligned}\psi^s_{[\mu]}[k] &\stackrel{\text{def}}{=} \textstyle\sum_{l=0}^{N/2^{\mu-1}-1} h^s_{[\mu-1]}[l]\,\psi^0_{[\mu-1]}[k-2^{\mu-1}l],\\ \tilde{\psi}^s_{[\mu]}[k] &\stackrel{\text{def}}{=} \textstyle\sum_{l=0}^{N/2^{\mu-1}-1} \tilde{h}^s_{[\mu-1]}[l]\,\tilde{\psi}^0_{[\mu-1]}[k-2^{\mu-1}l],\ \ s=0,\dots,S-1.\end{aligned} \tag{17.12}$$

The m-level framelet transform of a signal $\mathbf{x}$ results in the set of coefficients

$$\begin{aligned}\mathbf{x} \longrightarrow \mathbf{Y}_m \stackrel{\text{def}}{=} &\left\{y^0_{[m]}[l]\right\} \cup \left\{y^1_{[m]}[l]\right\} \cup \left\{y^2_{[m]}[l]\right\}\\ \cup \left\{y^1_{[m-1]}[l]\right\} &\cup \left\{y^2_{[m-1]}[l]\right\} \cup \cdots \cup \left\{y^1_{[1]}[l]\right\} \cup \left\{y^2_{[1]}[l]\right\}, \quad l \in \mathbb{Z}.\end{aligned} \tag{17.13}$$

The transform is invertible. Once the set $\mathbf{Y}_m$ of coefficients is available, consecutive application of the operations Eq. (17.11) restores the low-frequency coefficients. In

the end, Eq. (17.8) restores the signal $\mathbf{x}$. Diagrams in Fig. 17.1 illustrate direct and inverse multilevel framelet transforms with a three-channel p-filter bank.

17.1.2.3 Implementation of Multi-Level Frame Transform

For uniformity, let's denote, temporarily, the PR pair of p-filter banks $\tilde{\mathbf{H}}$ and $\mathbf{H}$ from the initial level as $\tilde{\mathbf{H}}_{[0]}$ and $\mathbf{H}_{[0]}$, respectively. Their polyphase matrices are denoted as $\tilde{\mathbf{P}}_{[0]}[n]$ and $\mathbf{P}_{[0]}[n]$. Assume that $\mathbf{x} = \{x[k]\} \in \mathit{\Pi}[N]$ is a signal to be operated by the m-level frame transform with the PR pairs $\tilde{\mathbf{H}}_{[\mu]}$ and $\mathbf{H}_{[\mu]}$, $\mu = 0, \dots, m-1$. For simplicity, assume that the polyphase matrices of the p-filter banks from the adjacent levels are linked as $\tilde{\mathbf{P}}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \tilde{\mathbf{P}}_{[\mu]}[2n]$, $\mathbf{P}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \mathbf{P}_{[\mu]}[2n]$.

Algorithm 3 Direct transform

1: Initialization: $\mathbf{y} \stackrel{\text{def}}{=} \mathbf{x}$. Split $\mathbf{y}$ into the even and the odd polyphase components $\mathbf{y} = \mathbf{y}_0 + \mathbf{y}_1$ and compute their DFT: $\{\hat{y}_0[n]_1\}$ and $\{\hat{y}_1[n]_1\}$.

2: For $\mu = 1$ to m Implement the one-level transform

$$\begin{pmatrix} \hat{y}^0_{[\mu]}[n]_\mu \\ \vdots \\ \hat{y}^{S-1}_{[\mu]}[n]_\mu \end{pmatrix} = \tilde{\mathbf{P}}_{[\mu-1]}[-n] \cdot \begin{pmatrix} \hat{y}_0[n]_\mu \\ \hat{y}_1[n]_\mu \end{pmatrix}$$

Compute the IDFT

$$y^s_{[\mu]}[k] = \frac{2^\mu}{N} \sum_{n=-N/2^{\mu+1}}^{N/2^{\mu+1}-1} \omega^{2^\mu kn}\, \hat{y}^s_{[\mu]}[n]_\mu, \quad s = 1, \dots, S-1,$$

and store the transform coefficients $\mathbf{y}^s_{[\mu]} \stackrel{\text{def}}{=} \left\{y^s_{[\mu]}[k]\right\}$, $k = 0, \dots, N/2^\mu - 1$, $s = 1, \dots, S-1$ of the μ-th decomposition level

Put $\hat{y}[n]_\mu \stackrel{\text{def}}{=} \hat{y}^0_{[\mu]}[n]_\mu$, $n \in \mathbb{Z}$.

Compute the DFT $\{\hat{y}_0[n]_{\mu+1}\}$ and $\{\hat{y}_1[n]_{\mu+1}\}$ of the even and the odd polyphase components of the signal $\mathbf{y}$:

$$\hat{y}_0[n]_{\mu+1} = \frac{\hat{y}[n]_\mu + \hat{y}[n + N/2^{\mu+1}]_\mu}{2},$$

$$\hat{y}_1[n]_{\mu+1} = \frac{\hat{y}[n]_\mu - \hat{y}[n + N/2^{\mu+1}]_\mu}{2\omega^{-2^\mu k}}$$

3: ———————————————————————-

Compute the IDFT

$$y^0_{[m]}[k] = \frac{2^m}{N} \sum_{n=-N/2^{m+1}}^{N/2^{m+1}-1} \omega^{2^m kn}\, \hat{y}^0_{[m]}[n]_m$$

and store the transform coefficients $\mathbf{y}^0_{[m]} \stackrel{\text{def}}{=} \left\{y^0_{[m]}[k]\right\}$, $k = 0, \dots, N/2^m - 1$

Algorithm 4 Inverse transform

Initialization: Compute the DFT

$$\hat{y}^s_{[m]}[n]_m = \sum_{k=0}^{N/2^m-1} \omega^{-2^m kn}\, y^s_{[m]}[k],\ k = 0, \ldots, N/2^m - 1, \quad s = 0, \ldots, S-1$$

For $\mu = m-1$ to 0 Implement the one-level inverse transform

$$\begin{pmatrix} \hat{y}_0[n]_\mu \\ \hat{y}_1[n]_\mu \end{pmatrix} = \mathbf{P}_{[\mu]}[n] \cdot \begin{pmatrix} \hat{y}^0_{[\mu+1]}[n]_{\mu+1} \\ \vdots \\ \hat{y}^{S-1}_{[\mu+1]}[n]_{\mu+1} \end{pmatrix}$$

Compute the DFT

$$\hat{y}^s_{[\mu]}[n]_\mu = \sum_{k=0}^{N/2^\mu-1} \omega^{-2^\mu kn}\, y^s_{[\mu]}[k],$$

$$k = 0, \ldots, N/2^\mu - 1, s = 1, \ldots, S-1$$

Compute the DFT

$$\hat{y}^0_{[\mu]}[n]_\mu = \hat{y}_0[n]_{\mu+1} + \omega^{-2^\mu kn} \hat{y}_1[n]_{\mu+1}$$

Compute the IDFT

$$y^0_{[0]}[k] = \frac{1}{N} \sum_{n=0}^{N/-1} \omega^{kn}\, \hat{y}^0_{[0]}[n]$$

,

Put $x[k] = y^0_{[0]}[k]$

17.2 Design of Interpolating Three-Channel p-Filter Banks Generating Frames

In this section, we discuss the design of three-channel linear phase p-filter banks, which generate tight and so-called semi-tight frames in the space $\Pi[N]$ of periodic signals.

17.2.1 Interpolating p-Filter Banks for Frame Generation

Assume that $\tilde{\mathbf{H}} = \left\{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1, \tilde{\mathbf{h}}^2\right\}$ and $\mathbf{H} = \left\{\mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2\right\}$ are the analysis and the synthesis p-filter banks that operate in the space $\Pi[N]$. Assume that the low-pass

p-filters $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^0$ are interpolating, that is their frequency responses can be represented as

$$\hat{\tilde{h}}^0[n] = \frac{1}{\sqrt{2}}\left(1 + \omega^{-n}\hat{\tilde{f}}[n]_1\right), \quad \hat{h}^0[n] = \frac{1}{\sqrt{2}}\left(1 + \omega^{-n}\hat{f}[n]_1\right),$$

where $\hat{\tilde{f}}[n]_1$ and $\hat{f}[n]_1$ are $N/2-$periodic sequences, which are the frequency responses of the low-pass p-filters $\tilde{\mathbf{f}}$ and $\mathbf{f}$, which operate in the space $\Pi[N/2]$. Denote $W[n]_1 \stackrel{\text{def}}{=} 1 - \hat{\tilde{f}}[-n]_1\,\hat{f}[n]_1$.

We assume that the following conditions are satisfied:

Conditions:

A1. $\hat{f}[n]_1$ and $\hat{\tilde{f}}[n]_1$ are rational functions of $\omega^n = e^{2\pi i n/N}$ that have no poles for $n \in \mathbb{Z}$.

A2. $\hat{f}[0]_1 = \hat{\tilde{f}}[0]_1 = 1$.

A3. Symmetry: $\omega^{-n}\hat{\tilde{f}}[n]_1 = \omega^n\,\hat{\tilde{f}}[-n]_1, \quad \omega^{-n}\hat{f}[n]_1 = \omega^n\,\hat{f}[-n]_1$.

Then, the p-filters $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^0$ have linear phase. The polyphase matrices for filter banks comprising the interpolating low-pass filters are

$$\tilde{\mathbf{P}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} 1/\sqrt{2} & \hat{\tilde{f}}[n]_1/\sqrt{2} \\ \hat{\tilde{h}}_0^1[n]_1 & \hat{\tilde{h}}_1^1[n]_1 \\ \hat{\tilde{h}}_0^2[n]_1 & \hat{\tilde{h}}_1^2[n]_1 \end{pmatrix}, \quad \mathbf{P}[n] \stackrel{\text{def}}{=} \begin{pmatrix} 1/\sqrt{2} & \hat{h}_0^1[n]_1\ \hat{h}_0^2[n]_1 \\ \hat{f}[n]_1/\sqrt{2} & \hat{h}_1^1[n]_1\ \hat{h}_1^2[n]_1 \end{pmatrix}.$$

The PR condition Eq. (17.5) leads to

$$\mathbf{P}^{12}[n] \cdot\ \tilde{\mathbf{P}}^{12}[-n] = \mathbf{Q}[n], \tag{17.14}$$

where

$$\tilde{\mathbf{P}}^{12}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^1[n]_1 & \hat{\tilde{h}}_1^1[n]_1 \\ \hat{\tilde{h}}_0^2[n]_1 & \hat{\tilde{h}}_1^2[n]_1 \end{pmatrix}, \quad \mathbf{P}^{12}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^1[n]_1 & \hat{h}_0^2[n]_1 \\ \hat{h}_1^1[n]_1 & \hat{h}_1^2[n]_1 \end{pmatrix},$$

$$\mathbf{Q}[n] \stackrel{\text{def}}{=} \frac{1}{2}\begin{pmatrix} 1 & -\hat{\tilde{f}}[-n]_1 \\ -\hat{f}[n]_1 & 2 - \hat{f}[n]_1\,\hat{\tilde{f}}[-n]_1 \end{pmatrix}.$$

Therefore, design of a three-channel PR p-filter bank starting from the interpolating low-pass filters $\tilde{\mathbf{h}}^0$ and $\mathbf{h}^0$ reduces to factorization of the matrix $\mathbf{Q}[n]$ as in Eq. (17.14).

A straightforward option is the triangular factorization of $\mathbf{Q}[n]$:

$$\tilde{\mathbf{P}}^{12}[n] = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & \tilde{w}[n]_1 \\ 1 & -\hat{\tilde{f}}[n]_1 \end{pmatrix}, \quad \mathbf{P}^{12}[n] = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 \\ w[n]_1 & -f[n]_1 \end{pmatrix},$$

where $w[n]_1\, \tilde{w}[-n]_1 = 2W[n]_1$.

Thus, to complete the design, the function $W[n]_1$ should be factorized. As soon as it is done, the PR p-filter bank $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$, where $\tilde{\mathbf{H}} = \left\{\tilde{\mathbf{h}}^0,\ \tilde{\mathbf{h}}^1,\ \tilde{\mathbf{h}}^2\right\}$, and $\mathbf{H} = \left\{\mathbf{h}^0,\ \mathbf{h}^1,\ \mathbf{h}^2\right\}$, whose frequency responses are

$$\begin{aligned} &\hat{\tilde{h}}^0[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(1 + \omega^{-n} \hat{\tilde{f}}[n]_1\right), \quad \hat{h}^0[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(1 + \omega^{-n} f[n]_1\right), \\ &\hat{\tilde{h}}^1[n] = \omega^{-n} \tilde{w}[n]_1/\sqrt{2}, \quad \hat{h}^1[n] = \omega^{-n} w[n]_1/\sqrt{2} \\ &\hat{\tilde{h}}^2[n] = \frac{1 - \omega^{-n} \hat{\tilde{f}}[n]_1}{\sqrt{2}} = \hat{\tilde{h}}^0[n + N/2], \quad \hat{h}^2[n] = \frac{1 - \omega^{-n} f[n]_1}{\sqrt{2}} = \hat{h}^0[n + N/2], \end{aligned} \tag{17.15}$$

comes into existence.

In this case, the p-filters $\mathbf{h}^2$, and $\tilde{\mathbf{h}}^2$ are interpolating as well as the p-filters $\mathbf{h}^0$ and $\tilde{\mathbf{h}}^0$. The condition $f[0] = \tilde{f}[0] = 1$ implies that

$$\tilde{h}^2[0] = h^2[0] = 0, \quad \tilde{h}^2[N/2] = h^2[N/2] = \sqrt{1},$$

and these p-filters are high-pass. The p-filters $\tilde{\mathbf{h}}^1$ and $\mathbf{h}^1$ do not have even polyphase components. The PR p-filter bank Eq. (17.15) generates a bi-frame in the periodic signals space $\Pi[N]$. The framelets of the first decomposition level are equal to the IRs of the corresponding filters: $\tilde{\psi}^s_{[1]}[k] = \tilde{h}^s[k]$, $\psi^s_{[1]}[k] = h^s[k]$, $s = 0, 1, 2,\ k \in \mathbb{Z}$. The framelets of the subsequent decomposition levels can be derived by iterated application of the formulas in Eq. (17.12).

However, a simpler way to derive, for example, a synthesis framelet $\psi^s_{[m]}$ of the level μ is to launch the m-level inverse transform using the polyphase matrices $\mathbf{P}_{[\mu]}$ as it is described in Sect. 17.1.2.3, provided that all the transform coefficients except for the single coefficient $y^s_{[\mu]}[1]$ are zero and $y^s_{[\mu]}[1] = 1$. In order to derive an analysis framelet $\tilde{\psi}^s_{[\mu]}$, one can apply the same procedure, using the analysis polyphase matrices $\tilde{\mathbf{P}}_{[\mu]}$ instead of the synthesis matrices $\mathbf{P}_{[\mu]}$. This is justified by the fact that synthesis and analysis frames are interchangeable.

17.2.2 Tight and Semi-Tight Frames

When the sequence $\hat{f}[n]_1$ is equal to $\hat{\tilde{f}}[n]_1$, we have

$$\begin{aligned} \hat{h}^0[n] &= \hat{\tilde{h}}^0[n], \quad \hat{h}^2[n] = \hat{\tilde{h}}^2[n] = h^0[n + N/2], \\ W[n]_1 &= 1 - |\hat{f}[n]_1|^2 = 2\hat{h}^0[n]\,\hat{h}^2[n]. \end{aligned} \tag{17.16}$$

Condition **A1** and the symmetry condition **A3** imply that, in this case, the numerator and the denominator of the sequence $W[n]_1$ are trigonometric polynomials with real coefficients containing only cosines of the form $\cos 2\pi\nu n/N$.
Tight frames: If the inequality

$$|\hat{f}[n]_1| \le 1 \ \text{ as } n \in \mathbb{Z} \tag{17.17}$$

holds, then, the sequence $W[n]_1$ can be factored as $W[n]_1 = w[n]_1\, w[-n]_1/2$. This factorization is not unique. However, the following Riesz's lemma holds.

Lemma 17.1 *[16] Let $A(z)$ be a positive trigonometric polynomial containing only cosines, $A(z) = \sum_{\nu=1}^{M} a_\nu \cos \nu z$, with real coefficients a_ν. Then, there exists a trigonometric polynomial $B(z) = \sum_{\nu=1}^{M} b_\nu\, e^{i\nu z}$ order* M *with real coefficients b_ν such that $|B(z)|^2 = A(z)$.*

Thus, a rational factorization $2W[n]_1 = w[n]_1\, w[-n]_1$ is possible and we have $h^1[n] = \tilde{h}^1[n]$. The analysis p-filter bank coincides with the synthesis p-filter bank

$$\begin{aligned} \hat{h}^0[n] &= \hat{\tilde{h}}^0[n] = (1 + \omega^{-n}\hat{f}[n]_1)/\sqrt{2}, \\ \hat{h}^2[n] &= \hat{\tilde{h}}^2[n] = (1 - \omega^{-n}\hat{f}[n]_1)/\sqrt{2}, \\ \hat{h}^1[n] &= \hat{\tilde{h}}^1[n] = \omega^{-n}\, w[n]_1/\sqrt{2}, \end{aligned} \tag{17.18}$$

that generates a tight frame. Note that symmetry of the sequence $W[n]_1$ does not guarantee (anti)symmetry of the sequence $w[n]_1 = \sqrt{2\,W[n]_1}$.
Semi-tight frames: If the condition in Eq. (17.17) is not satisfied, it is still possible to generate frames that are very close to tight frames. In this case, the sequence $W[n]_1$ can be factored as $2W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$ and the frequency responses of the p-filter bank are

$$\begin{aligned} \hat{h}^0[n] &= \hat{\tilde{h}}^0[n] = (1 + \omega^{-n}\hat{f}[n]_1)/\sqrt{2}, \\ \hat{h}^2[n] &= \hat{\tilde{h}}^2[n] = (1 - \omega^{-n}\hat{f}[n]_1)/\sqrt{2}, \\ \hat{\tilde{h}}^1[n] &= \omega^{-n}\tilde{w}[n]_1/\sqrt{2}, \quad \hat{h}^1[n] = \omega^{-n} w[n]_1/\sqrt{2}. \end{aligned} \tag{17.19}$$

Such a p-filter bank generates a frame, which is natural to name the *semi-tight* frame. In non-periodic setting, this notion was introduced in [1]. Due to the symmetry of $W[n]_1$, its (anti)symmetric factorization $2W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$ is always pos-

sible. Therefore, even when Eq. (17.17) holds, sometimes it is preferable to use a semi-tight rather than a tight frame.

17.2.3 Interpolating Three-Channel p-Filter Banks Using Spline Filters

In this section, we present a few examples of PR three-channel p-filter banks, which generate tight and semi-tight frames in the space $\Pi[N]$ of N-periodic discrete-time signals. In all the forthcoming examples, the low-pass and the high-pass p-filters are interpolating, which utilize spline-based prediction p-filters. These prediction p-filters are introduced in Sects. 16.2 and 16.3. The design scheme is described in Sect. 17.2.2.

17.2.3.1 p-Filter Banks Originating From Polynomial Splines

Linear polynomial spline:

The frequency response of the prediction p-filter is $\hat{f}_c^2[n]_1 = (1+\omega^{2n})/2$. Thus, the sequence

$$2W[n]_1 \stackrel{\text{def}}{=} 2(1 - \hat{f}_c^2[n]_1\, \hat{f}_c^2[-n]) = \frac{-\omega^{2n} + 2 - \omega^{-2n}}{2}$$

can be factorized to be $2W[n]_1 = w[n]_1\, w[-n]_1$, where $w[n]_1 \stackrel{\text{def}}{=} (1-\omega^{2n})/\sqrt{2}$. The analysis p-filter bank, which coincides with the synthesis p-filter bank $\mathbf{H} = \{\mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2\}$, generates a tight frame in $\Pi[N]$. The frequency responses of the p-filters are

$$\begin{aligned}
\hat{h}^0[n] &= \frac{1}{\sqrt{2}}\left(1 + \omega^{-n}\, \hat{f}_c^2[n]_1\right) = \sqrt{2}\, \cos^2 \frac{\pi n}{N}, \\
\hat{h}^1[n] &= \frac{\omega^{-n}\, w[n]_1}{\sqrt{2}} = \frac{\omega^{-n} - \omega^n}{2}, \\
\hat{h}^2[n] &= \frac{1}{\sqrt{2}}\left(1 - \omega^{-n}\, \hat{f}_c^2[n]_1\right) = \sqrt{2}\, \sin^2 \frac{\pi n}{N}.
\end{aligned} \tag{17.20}$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of the first degree while the p-filter $\mathbf{h}^2$ locally eliminates them. The p-filter $\mathbf{h}^1$ eliminates only constants. Thus, the framelets $\psi^1_{[1]}$ and $\psi^2_{[1]}$, which are the IRs of the p-filters $\mathbf{h}^1$ and $\mathbf{h}^2$, respectively, have one and two LDVMs, respectively. The impulse and the magnitude responses of the filters $\mathbf{h}^s$ are displayed in Fig. 17.4.

Quadratic interpolating spline: $p = 3$

Denote $\Omega_c^3[n]_1 \stackrel{\text{def}}{=} \cos^4 \pi n/N + \sin^4 \pi n/N$. Then,

$$\hat{f}_c^3[n]_1 = \omega^n \frac{\cos^4 \pi n/N - \sin^4 \pi n/N}{\Omega_c^3[n]_1}, \quad W[n]_1 = \left(\frac{\sin^2 2\pi n/N}{2\Omega_c^3[n]_1}\right)^2 \tag{17.21}$$

$$\hat{h}^0[n] = \sqrt{2}\,\frac{\cos^4 \pi n/N}{\Omega_c^3[n]_1}, \quad \hat{h}^2[n] = \sqrt{2}\,\frac{\sin^4 \pi n/N}{\Omega_c^3[n]_1} \quad \hat{h}^1[n] = \frac{\omega^{-n}}{2}\,\frac{\sin^2 2\pi n/N}{\Omega_c^3[n]_1}.$$

The p-filter $\mathbf{h}^0$ locally restores sampled cubic polynomials while the p-filter $\mathbf{h}^2$ locally eliminates them. The p-filter $\mathbf{h}^1$ eliminates first degree polynomials. Thus, the framelets $\psi^1_{[1]}$ and $\psi^2_{[1]}$ have two and four LDVMs, respectively. The impulse and the magnitude responses of the filters $\mathbf{h}^s$ are displayed in Fig. 17.4.

Cubic interpolating spline: $p = 4$:

Denote

$$\Omega_c^4[n]_1 \stackrel{\text{def}}{=} \omega^{2n} + 4 + \omega^{-2n}, \quad \Gamma_c^4[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 14 - \omega^{-2n}. \tag{17.22}$$

Then, we have

$$\begin{aligned}
\hat{f}_c^4[n]_1 &= \frac{\omega^{4n} + 23\omega^{2n} + 23 + \omega^{-2n}}{8\Omega_c^4[n]_1}, \\
W[n]_1 &= \left(\frac{\omega^{2n} - 2 + \omega^{-2n}}{8\Omega_c^4[n]_1}\right)^2 \Gamma_c^4[n]_1, \\
\hat{h}^0[n] &= \sqrt{2}\,\frac{\cos^4 \pi n/N\ (2 + \cos 2\pi n/N)}{\Omega_c^4[n]_1}, \\
\hat{h}^2[n] &= \sqrt{2}\,\frac{\sin^4 \pi n/N\ (2 - \cos 2\pi n/N)}{\Omega_c^4[n]_1}.
\end{aligned} \tag{17.23}$$

Factorization $2W[n]_1 = w[n]_1\, w[-n]_1$, which results in a tight frame, provides the p-filter $\mathbf{h}^1$, whose frequency response is

$$\hat{h}^1[n] = \omega^{-n}\,\frac{\omega^{2n} - 2 + \omega^{-2n}}{8\sqrt{q}\ \Omega_c^4[n]_1}\left(1 - q\omega^{2n}\right), \tag{17.24}$$

where $q = 7 - 4\sqrt{3} \approx 0.0718$. The frequency response is not symmetric about swap $n \to -n$ and, consequently, the framelet $\psi^1_{[1]}$ is slightly asymmetric. It has two LDVMs, whereas the framelet $\psi^2_{[1]}$ has four LDVMs. Figure 17.4 displays the

IRs of the filters $\mathbf{h}^s$, which are the discrete time framelets of the first level, and their magnitude responses. Observe that the IR of the filter $\mathbf{h}^1$ is non-symmetric.

Framelets symmetry can be achieved by factorization $2W[n]_1 = w[n]_1 \, \tilde{w}[-n]_1$ with the unequal factors $w[n]_1$ and $\tilde{w}[n]_1$, which leads to a semi-tight frame with $\tilde{\psi}^1_{[1]} \neq \psi^1_{[1]}$. Non-uniqueness of such a factorization provides additional adaptation abilities. For example, either the analysis $\tilde{\mathbf{h}}^1$ or the synthesis $\mathbf{h}^1$ p-filter can be chosen to be p-FIR. The local discrete vanishing moments can be transferred from the synthesis $\psi^1_{[1]}$ to the analysis $\tilde{\psi}^1_{[1]}$ framelet and vice versa.

The following are examples of the p-filters $\tilde{\mathbf{h}}^1$ and $\mathbf{h}^1$ that result from three different factorization modes.

1. *Symmetric factorization* with equal number (two) of LDVMs in the analysis $\tilde{\psi}^1_{[1]}$ and synthesis $\psi^1_{[1]}$ framelets:

$$\begin{aligned} \hat{\tilde{h}}^1[n] &= \omega^{-n} \frac{\omega^{2n} - 2 + \omega^{-2n}}{2\, \Omega_c^4[n]_1}, \\ \hat{h}^1[n] &= \omega^{-n} \frac{\left(\omega^{2n} - 2 + \omega^{-2n}\right)\ \Gamma_c^4[n]_1}{32\, \Omega_c^4[n]_1}. \end{aligned} \tag{17.25}$$

2. *Symmetric factorization* with equal number (two) of LDVMs in the analysis $\tilde{\psi}^1_{[1]}$ and synthesis $\psi^1_{[1]}$ framelets. The analysis p-filter $\tilde{\mathbf{h}}^1$ is p-FIR and, consequently, the framelet $\tilde{\psi}^1_{[1]}$ is compactly supported:

$$\begin{aligned} \hat{\tilde{h}}^1[n] &= \omega^{-n} \frac{\omega^{2n} - 2 + \omega^{-2n}}{4}, \\ \hat{h}^1[n] &= \omega^{-n} \frac{\left(\omega^{2n} - 2 + \omega^{-2n}\right)\ \Gamma_c^4[n]_1}{16\ \left(\Omega_c^4[n]_1\right)^2}. \end{aligned} \tag{17.26}$$

3. *Antisymmetric factorization* with three LDVMs in the analysis $\tilde{\psi}^1_{[1]}$ and one LDVM in synthesis $\psi^1_{[1]}$ framelets:

$$\begin{aligned} \hat{\tilde{h}}^1[n] &= 3\omega^{-n} \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{4\, \Omega_c^4[n]_1}, \\ \hat{h}^1[n] &= \omega^{-n} \frac{\left(1 - \omega^{2n}\right)\ \Gamma_c^4[n]_1}{48\, \Omega_c^4[n]_1}. \end{aligned} \tag{17.27}$$

Figure 17.2 displays the IRs and the magnitude responses of the p-filters $\tilde{\mathbf{h}}^1$ and $\mathbf{h}^1$, which stem from different factorizations of the sequence $W[n]_1$ given in Eq. (17.23). The IRs of the p-filters $\mathbf{h}^1$ are either symmetric or antisymmetric.

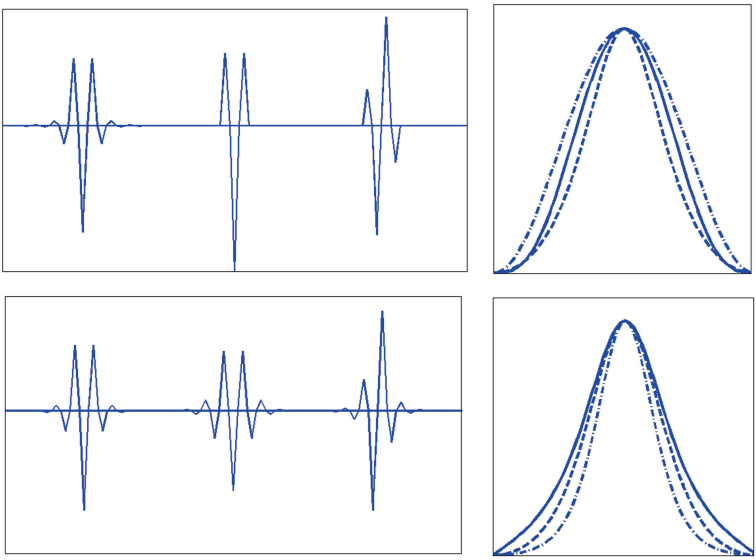


Fig. 17.2 *Top left* IRs of the analysis p-filters $\mathbf{h}^1$ for the semi-tight frames, which are defined in Eqs. *left* to *right*: (17.25) $\longrightarrow$ (17.26) $\longrightarrow$ (17.27). *Bottom left* IRs of the corresponding analysis p-filters $\tilde{\mathbf{h}}^1$. *Top right* Magnitude responses of the synthesis p-filters $\mathbf{h}^1$, which are defined in Eqs. (17.25) (*dashed line*), (17.26) (*dash-dot line*) and (17.27) (*solid line*). *Bottom right*: Magnitude responses of the corresponding analysis p-filters $\tilde{\mathbf{h}}^1$

Quadratic quasi-interpolating spline:

In this case, all the p-filters have finite IRs (up to periodization). Equation (16.20) implies that

$$f_q^3[n]_1 = \frac{-\omega^{-2n} + 9 + 9\omega^{2n} - \omega^{4n}}{16}, \tag{17.28}$$
$$W[n]_1 = \left(\frac{\omega^{2n} - 2 + \omega^{-2n}}{16}\right)^2 \Gamma_c^4[n]_1,$$

where the sequence $\Gamma_c^4[n]_1$ is defined in Eq. (17.60). The frequency responses of the low- and the high-pass p-filters are

$$\hat{h}^0[n] = \sqrt{2}\,\cos^4\frac{\pi n}{N}\left(1 + 2\sin^2\frac{\pi n}{N}\right), \tag{17.29}$$
$$\hat{h}^2[n] = \sqrt{2}\,\sin^4\frac{\pi n}{N}\left(1 + 2\cos^2\frac{\pi n}{N}\right),$$

respectively. The following factorization modes $2W[n]_1 = w[n]_1\,\tilde{w}[-n]_1$ are similar to the previous modes for the cubic interpolating spline:

1. *Non-symmetric factorization* $2W[n]_1 = w[n]_1\, w[-n]_1$, which results in the frequency responses

$$\hat{\tilde{h}}^1[n] = \hat{h}^1[n] = \omega^{-n}\,\frac{(\omega^{-2n} - 2 + \omega^{2n})(1 - q\,\omega^{2n})}{16\sqrt{q}}, \quad q = 7 - 4\sqrt{3}. \tag{17.30}$$

The p-filter bank generates a tight frame. The framelet $\psi^1_{[1]}$ has two LDVMs, while $\psi^2_{[1]}$ has four.
Figure 17.9 displays the IRs of the p-filters $\mathbf{h}^s$, which are discrete-time framelets of the first level, and their magnitude responses. The IR of the p-filter $\mathbf{h}^1$ is not symmetric.

2. *Symmetric factorization* with equal number (two) of LDVMs in the analysis $\tilde{\psi}^1_{[1]}$ and in the synthesis $\psi^1_{[1]}$ framelets:

$$\begin{aligned}\hat{\tilde{h}}^1[n] &= \omega^{-n}\,\frac{\left(\omega^{-2n} - 2 + \omega^{2n}\right)}{4}, \\ \hat{h}^1[n] &= \omega^{-n}\,\frac{\left(\omega^{-2n} - 2 + \omega^{2n}\right)\ \Gamma_c^4[n]_1}{64}.\end{aligned} \tag{17.31}$$

3. *Antisymmetric factorization* with three LDVMs in the analysis $\tilde{\psi}^1(t)$ and one LDVM in the synthesis $\psi^1(t)$ framelets:

$$\begin{aligned}\hat{\tilde{h}}^1[n] &= \omega^{-n}\,\frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \\ \hat{h}^1[n] &= \omega^{-n}\,\frac{\left(1 - \omega^{2n}\right)\ \Gamma_c^4[n]_1}{32}.\end{aligned} \tag{17.32}$$

4. *Symmetric factorization* with all four LDVMs in the analysis $\tilde{\psi}^1(t)$ and none LDVM in the synthesis $\psi^1(t)$ framelets:

$$\begin{aligned}\hat{\tilde{h}}^1[n] &= \omega^{-n}\,\frac{\omega^{4n} - 4\omega^{2n} + 6 - 4\omega^{-2n} + \omega^{-4n}}{16}, \\ \hat{h}^1[n] &= \omega^{-n}\,\frac{\Gamma_c^4[n]_1}{16}.\end{aligned} \tag{17.33}$$

The p-filter $\mathbf{h}^1$ is all-pass.

Figure 17.3 displays the IRs and the magnitude responses of the p-filters $\tilde{\mathbf{h}}^1$ and $\mathbf{h}^1$, respectively. These stem from different factorizations of the sequence $W[n]$ given in Eq. (17.28). The IRs of the p-filters are either symmetric or antisymmetric.

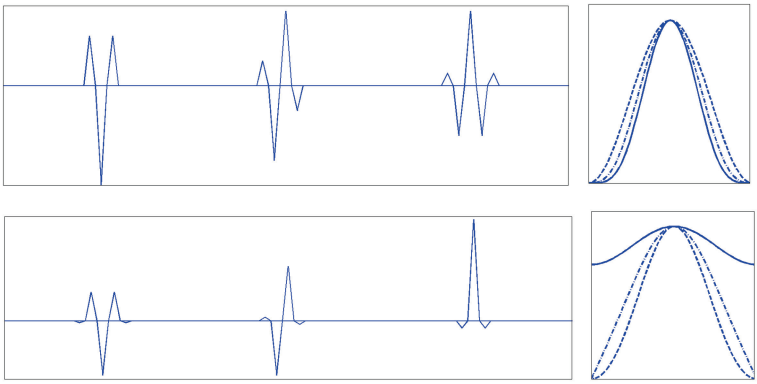


Fig. 17.3 *Top left* IRs of the analysis p-filters $\mathbf{h}^1$ for the semi-tight frames, which are defined in Eqs. *left* to *right*: (17.31) $\longrightarrow$ (17.32) $\longrightarrow$ (17.33). *Bottom left* IRs of the corresponding analysis p-filters $\tilde{\mathbf{h}}^1$. *Top right* Magnitude responses of the synthesis p-filters $\mathbf{h}^1$, which are defined in Eqs. (17.31) (*dashed line*), (17.32) (*dash-dot line*) and (17.33) (*solid line*). *Bottom right* Magnitude responses of the corresponding analysis p-filters $\tilde{\mathbf{h}}^1$

17.2.3.2 p-Filter Banks Derived From Discrete splines

Denote $\Omega_d^{2r}[n]_1 \stackrel{\text{def}}{=} \cos^{2r} \pi n/N + \sin^{2r} \pi n/N$. Once the prediction p-filter is derived from a discrete spline of order $2r$, Eqs. (16.23), (2.4) and (2.5) imply that

$$\hat{f}_d^{2r}[n]_1 = \frac{\omega^n \left(\cos^{2r} \pi n/N - \sin^{2r} \pi n/N\right)}{\Omega_d^{2r}[n]_1}, \tag{17.34}$$

$$\hat{h}^0[n] \stackrel{\text{def}}{=} \frac{\sqrt{2}\cos^{2r} \pi n/N}{\Omega_d^{2r}[n]_1}, \quad \hat{h}^2[n] \stackrel{\text{def}}{=} \frac{\sqrt{2}\sin^{2r} \pi n/N}{\Omega_d^{2r}[n]_1}. \tag{17.35}$$

The low-pass $\mathbf{h}^0$ and the high-pass $\mathbf{h}^2$ p-filters locally restore and eliminate sampled polynomials of degree $2r-1$, respectively, thus the framelet $\psi^2_{[1]}$ has $2r$ LDVM. The sequence $W[n]_1$

$$2W[n]_1 = \frac{8\,\sin^{2r} 2\pi n/N}{2^{2r}\left(\Omega_d^{2r}[n]_1\right)^2} = \frac{8(-1)^r\,\omega^{-2nr}\left(1-\omega^{2n}\right)^{2r}}{4^{2r}\left(\Omega_d^{2r}[n]_1\right)^2} = w[n]_1\,w[-n]_1, \tag{17.36}$$

$$w[n]_1 = \frac{\sqrt{8}\,\omega^{-2np}\left(1-\omega^{2n}\right)^r}{4^r\,\Omega_d^{2r}[n]_1}\ (r \text{ is odd}), \quad w[n]_1 = \frac{\sqrt{8}\,\sin^r 2\pi n/N}{2^r\,\Omega_d^{2r}[n]_1}\ (r \text{ is even}).$$

The band-pass p-filter $\mathbf{h}^1$, whose frequency response is $h^1[n] = \omega^{-n} w[n]_1/\sqrt{2}$, eliminates sampled polynomials of degree $r-1$, thus, the framelet $\psi^1_{[1]}$ has r LDVM.

The triple $\{\mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2\}$ forms a PR p-filter bank, which generates a tight frame in the space $\Pi[N]$.

Discrete spline of fourth order: $r = 2$

It is readily verified that, in this case, the p-filters coincide with the p-filters derived from the quadratic interpolating polynomial spline.

Discrete spline of sixth order: $r = 3$

In this case,

$$\hat{h}^1[n] = \frac{\omega^{-n}}{2} \frac{\omega^{-2n}\left(1-\omega^{2n}\right)^3}{3\,\omega^{2n}+10+3\,\omega^{-2n}} = \frac{\omega^{-n}}{2} \frac{\omega^{-2n}-3+3\omega^{2n}-\omega^{4n}}{3\,\omega^{2n}+10+3\,\omega^{-2n}}. \quad (17.37)$$

The p-filter $\mathbf{h}^1$ locally eliminates sampled quadratic polynomials. Thus, the framelet $\psi^1_{[1]}$ is antisymmetric and has two LDVM. The impulse and the magnitude responses of the filters $\mathbf{h}^s$ are displayed in Fig. 17.4.

Discrete spline of eighth order: $r = 4$.

In this case,

$$\hat{h}^1[n] = \frac{\omega^{-n}}{2} \frac{\left(\omega^{2n}-2+\omega^{-2n}\right)^2}{\omega^{4n}+28\,\omega^{2n}+70+28\,\omega^{-2n}+\omega^{-4n}}. \quad (17.38)$$

The p-filter $\mathbf{h}^1$ locally eliminates sampled cubic polynomials. Thus, the framelet $\psi^1_{[1]}$ is symmetric and has four LDVM. The impulse and magnitude responses of the filters $\mathbf{h}^s$ are displayed in Fig. 17.4.

Remark 17.2 *One of advantages of the semi-tight frames is the option to swap the LDVM between the analysis and the synthesis framelets. Typically, it is preferable to have more LDVM in analysis rather than in synthesis framelets even in cases when the tight frame p-filters have linear phase.*

As an example, we introduce a semi-tight frame originated from the quadratic interpolating spline, where the frequency response of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^2$ are given in Eq. (17.21) and

$$\hat{\tilde{h}}^1[n] = \frac{\omega^{-n}}{2} \frac{\sin^4 2\pi n/N}{\Omega_c^3[n]_1}, \quad \hat{h}^1[n] = \frac{\omega^{-n}}{2} \frac{1}{\Omega_c^3[n]_1}, \quad (17.39)$$

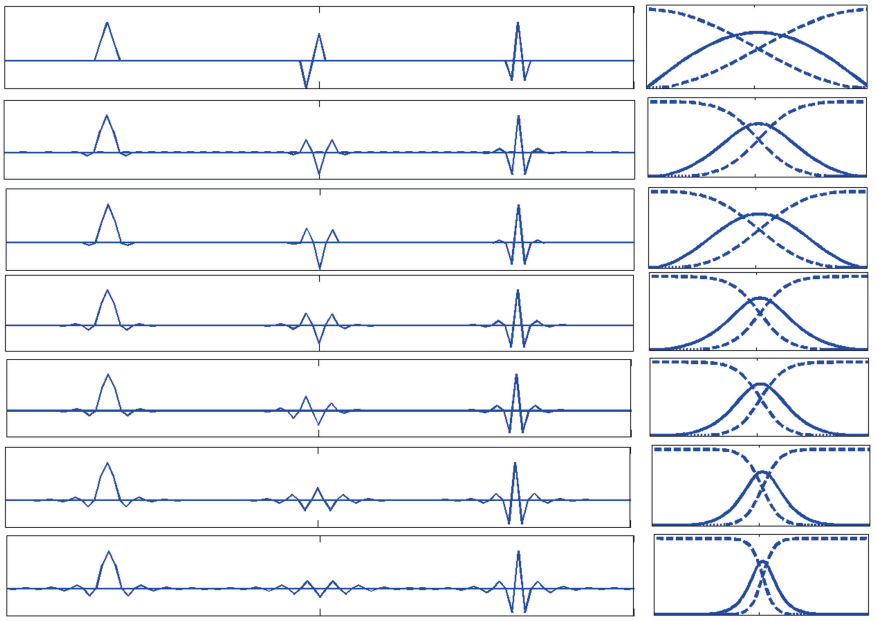


Fig. 17.4 Impulse and magnitude responses of the three-channel p-filter banks that generate tight frames. *Left* pictures display the impulse responses of p-filters where, from *left* to *right*, we have: $\mathbf{h}^0 \longrightarrow \mathbf{h}^1 \longrightarrow \mathbf{h}^2$. *Right* pictures display the magnitude responses of these p-filters: $\mathbf{h}^0$ and $\mathbf{h}^2$ (*dashed lines*), and $\mathbf{h}^1$ (*solid line*). *Top*: linear interpolating spline. Second from *top*: Quadratic interpolating spline(discrete spline of order 4). Third from *top*: Quadratic quasi-interpolating spline. *Center*: Cubic interpolating spline. Third from the *bottom*: discrete spline of order 6. Second from the *bottom*: discrete spline of order 8. *Bottom*: discrete spline of order 12

$\Omega_c^3[n]_1 \stackrel{\text{def}}{=} \cos^4 \pi n/N + \sin^4 \pi n/N$. The analysis framelet $\tilde{\psi}_{[1]}^1$ has four LDVM while the synthesis framelet $\psi_{[1]}^1$ has none.

One more example is the semi-tight frame originating from the sixth order discrete spline, where the frequency response of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^2$ are given in Eq. (17.35) and

$$\hat{\tilde{h}}^1[n] = \frac{\omega^{-n}}{2} \frac{\sin^4 2\pi n/N}{\Omega_d^6[n]_1}, \quad \hat{h}^1[n] = \frac{\omega^{-n}}{2} \frac{\sin^2 2\pi n/N}{\Omega_d^6[n]_1} \tag{17.40}$$

where $\Omega_d^6[n]_1 \stackrel{\text{def}}{=} \cos^6 \pi n/N + \sin^6 \pi n/N$. The analysis framelet $\tilde{\psi}_{[1]}^1$ has four LDVMs while the synthesis framelet $\psi_{[1]}^1$ has two. Figure 17.4 displays impulse responses of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2$, which are the discrete time framelets $\psi_{[1]}^s$ of the first level, and their magnitude responses. Frequency responses of the low-pass filters $\mathbf{h}^0$ and the corresponding high-pass filters $\mathbf{h}^2$ mirror each other.

Direct and inverse framelet transforms using three-channel p-filter banks are implemented by the MATLAB functions `fram34_dec_hm.m` and

`fram34_rec_hm.m`, respectively. When some framelet transforms are repeatedly applied, it is advisable to use the functions `fram34_dec_hmwf.m` and `fram34_rec_hmwf.m`. The two-dimensional transforms are implemented by the functions `fram34_dec2d_hm.m` and `fram34_rec2d_hm.m` (or `fram34_dec2d_hmwf.m` and `fram34_rec2d_hmwf.m`), respectively. Available frames are listed in the MATLAB code `list_frame3.m`. Figures 17.2 and 17.3 are produced by the MATLAB code `fram3_exam_P.m`. Figure 17.4 is produced by the MATLAB code `fram3_exam_P7.m`.

17.3 Design of Four-Channel p-Filter Banks for Frames Generation

In this section, we discuss design of four-channel p-filter banks, which generate tight and semi-tight frames in the space $\Pi[N]$ of periodic signals.

17.3.1 Four-Channel Perfect Reconstruction p-Filter Banks

The PR condition for a pair of the analysis $\tilde{\mathbf{H}} = \left\{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1, \tilde{\mathbf{h}}^2, \tilde{\mathbf{h}}^3\right\}$ and the synthesis $\mathbf{H} = \left\{\mathbf{h^0}, \mathbf{h^1}, \mathbf{h^2}, \mathbf{h^3}\right\}$ p-filter banks is expressed via their polyphase matrices

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_0^1[n]_1 & \hat{h}_0^2[n]_1 & \hat{h}_0^3[n]_1 \\ \hat{h}_1^0[n]_1 & \hat{h}_1^1[n]_1 & \hat{h}_1^2[n]_1 & \hat{h}_1^3[n]_1 \end{pmatrix} \cdot \begin{pmatrix} \hat{\tilde{h}}_0^0[-n]_1 & \hat{\tilde{h}}_1^0[-n]_1 \\ \hat{\tilde{h}}_0^1[-n]_1 & \hat{\tilde{h}}_1^1[-n]_1 \\ \hat{\tilde{h}}_0^2[-n]_1 & \hat{\tilde{h}}_1^2[-n]_1 \\ \hat{\tilde{h}}_0^3[-n]_1 & \hat{\tilde{h}}_1^3[-n]_1 \end{pmatrix} = \mathbf{I}_2, \tag{17.41}$$

$n \in \mathbb{Z}$. The matrix product in Eq. (17.41) can be split into two products:

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \mathbf{P}^{01}[n] \cdot \tilde{\mathbf{P}}^{01}[-n] + \mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{17.42}$$

$$\mathbf{P}^{01}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_0^1[n]_1 \\ \hat{h}_1^0[n]_1 & \hat{h}_1^1[n]_1 \end{pmatrix}, \quad \tilde{\mathbf{P}}^{01}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^0[n]_1 & \hat{\tilde{h}}_1^0[n]_1 \\ \hat{\tilde{h}}_0^1[n]_1 & \hat{\tilde{h}}_1^1[n]_1 \end{pmatrix}$$

$$\mathbf{P}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^2[n]_1 & \hat{h}_0^3[n]_1 \\ \hat{h}_1^2[n]_1 & \hat{h}_1^3[n]_1 \end{pmatrix} \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^2[n]_1 & \hat{\tilde{h}}_1^2[n]_1 \\ \hat{\tilde{h}}_0^3[n]_1 & \hat{\tilde{h}}_1^3[n]_1 \end{pmatrix}.$$

According to Theorem 17.1, a PR pair $\left\{\mathbf{H},\ \tilde{\mathbf{H}}\right\}$ of p-filter banks generate a tight frame if their polyphase matrices are linked as

$$\mathbf{P}[n] = \tilde{\mathbf{P}}[n]^T \Longleftrightarrow \mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T \quad \text{and} \quad \mathbf{P}^{23}[n] = \tilde{\mathbf{P}}^{23}[n]^T, \quad n \in \mathbb{Z}.$$

Definition 17.2 *Assume that* $\tilde{\mathbf{P}}[n]$ *and* $\mathbf{P}[n]$ *are polyphase matrices of the PR pair* $\left\{\tilde{\mathbf{H}},\ \mathbf{H}\right\}$ *of p-filter banks, which generate a bi-frame* $\left\{\tilde{\mathbf{F}},\ \mathbf{F}\right\}$ *in the space* $\Pi[N]$*. If the matrices* $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$ and $\mathbf{P}^{23}[n] \neq \tilde{\mathbf{P}}^{23}[n]^T$, $n \in \mathbb{Z}$, *then the frame* $\left\{\tilde{\mathbf{F}},\ \mathbf{F}\right\}$ *is called semi-tight.*

As in three-channel case, design of four-channel filter banks begins from a linear phase low-pass filter $\mathbf{h}^0 = \tilde{\mathbf{h}}^0$, whose FRs $\hat{h}^0[n] = \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_1^0[n]_1$ is a rational function of $\omega^n = e^{2\pi i n/N}$ with real coefficients that has no poles for $n \in \mathbb{Z}$. Assume that $\hat{h}^0[n]$ is symmetric about the swap $n \to -n$, which implies that $\hat{h}_0^0[n]_1 = \hat{h}_0^0[-n]_1$ and $\omega^{-n}\hat{h}_1^0[n]_1 = \omega^n\,\hat{h}_1^0[-n]_1$. The IR $\left\{h^0[k]\right\}$ is symmetric about $k = 0$.

In addition, assume that $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$ and the product

$$\mathbf{P}^{01}[n] \cdot \mathbf{P}^{01}[-n] = \begin{pmatrix} \alpha[n]_1 & 0 \\ 0 & \beta[n]_1 \end{pmatrix} \tag{17.43}$$

is a diagonal matrix. The assumption in Eq. (17.43) is equivalent to the condition $\hat{h}_0^0[n]_1\hat{h}_1^0[-n]_1 + \hat{h}_0^1[n]_1\hat{h}_1^1[-n]_1 = 0$. The simplest way to satisfy this condition is to define

$$\hat{h}_0^1[n]_1 \stackrel{\text{def}}{=} -\hat{h}_1^0[-n]_1,\ \hat{h}_1^1[n]_1 \stackrel{\text{def}}{=} \hat{h}_0^0[-n]_1 \Longrightarrow \alpha[n]_1 = \beta[n]_1 = \left|\hat{h}_0^0[n]_1\right|^2 + \left|\hat{h}_1^0[n]_1\right|^2.$$

Due to symmetry of $\hat{h}^0[n]$, the FR

$$\begin{aligned} \hat{\hat{h}}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1 \\ &= \omega^{-n}\left(\hat{h}_0^0[n]_1 - \omega^{-n}\hat{h}_1^0[n]_1\right) = \omega^{-n}\hat{h}^0[n + N/2]. \end{aligned} \tag{17.44}$$

Equation (17.44) implies that the sequence $\omega^n\hat{\hat{h}}^1[n]$ is symmetric about the swap $n \to -n$ and, consequently, the IR $\left\{h^1[k]\right\}$ is symmetric about $k = 1$.

It follows from the assumption in Eq. (17.43) that the product

$$\mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \mathbf{Q}[n] \stackrel{\text{def}}{=} \begin{pmatrix} t[n]_1 & 0 \\ 0 & t[n]_1 \end{pmatrix}, \quad \text{where} \tag{17.45}$$

$$t[n]_1 \stackrel{\text{def}}{=} 1 - \left|\hat{h}_0^0[n]_1\right|^2 + \left|\hat{h}_1^0[n]_1\right|^2,$$

is a diagonal matrix. Thus, design of the PR p-filter bank is reduced to factorization of the matrix $\mathbf{Q}[n]$.

17.3.1.1 Diagonal factorizAtion of the Matrix Q[*n*]

There are many ways to factorize the matrix $\mathbf{Q}[n]$. One way is to define the matrices $\mathbf{P}^{23}[n]$ and $\tilde{\mathbf{P}}^{23}[n]$ to be diagonal:

$$\mathbf{P}^{23}[n] = \begin{pmatrix} \hat{h}_0^2[n]_1 & 0 \\ 0 & \hat{h}_1^3[n]_1 \end{pmatrix}, \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{\tilde{h}}_0^2[n]_1 & 0 \\ 0 & \hat{\tilde{h}}_1^3[n]_1 \end{pmatrix},$$

which means that the odd polyphase components of the p-filters $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^2$ as well as the even polyphase components of the p-filters $\tilde{\mathbf{h}}^3$ and $\mathbf{h}^3$ vanish. Consequently, we have to find four sequences $\hat{h}_0^2[n]_1$, $\hat{\tilde{h}}_0^2[n]_1$, $\hat{h}_1^3[n]_1$ and $\hat{\tilde{h}}_1^3[n]_1$ such that

$$\hat{h}_0^2[n]_1 \, \hat{\tilde{h}}_0^2[-n]_1 = \hat{h}_1^3[n]_1 \, \hat{\tilde{h}}_1^3[-n]_1 = t[n]_1. \tag{17.46}$$

Tight frame p-filter banks:

If the following inequality holds

$$\alpha[n]_1 = \left| h_0^0[n]_1 \right|^2 + \left| h_1^0[n]_1 \right|^2 \le 1, \quad n \in \mathbb{Z}, \tag{17.47}$$

then, due to symmetry of the rational functions $\hat{h}_0^0[n]_1$ and $\hat{h^0}_0[n]_1$, we have $1 - a[n]_1 = P(\cos 2\pi n/N)/R(\cos 2\pi n/N)$, where R is strictly positive and P is non-negative polynomials. Due to the Riesz's Lemma 17.1, the polynomials can be factorized $P(\cos 2\pi n/N)) = p(\omega^n)\, p(\omega^{-n})$, $R(\cos 2\pi n/N)) = q(\omega^n)\, q(\omega^{-n})$, where p and q are polynomials with real coefficients and $q(\omega^n)$ does not have roots for $n \in \mathbb{Z}$. Thus, we can define

$$\hat{h}_0^2[n]_1 = \hat{\tilde{h}}_0^2[n]_1 = \hat{h}_1^3[-n]_1 = \hat{\tilde{h}}_1^3[-n]_1 = T[n]_1 \stackrel{\text{def}}{=} \frac{p(\omega^n)}{q(\omega^n)}. \tag{17.48}$$

The PR p-filter bank, whose FRs are

$$\begin{aligned} \hat{h}^0[n] &= \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_0^0[n]_1 & \hat{h}^2[n] &= T[n]_1, \\ \hat{h}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1, & \hat{h}^3[n] &= \omega^{-n}T[-n]_1, \end{aligned}$$

generates a tight wavelet frame in the space $\Pi[N]$. Certainly, symmetry of the FR $\hat{h}^0[n]$ does not guarantee symmetry of the FRs $\hat{h}^2[n]$ and $\hat{h}^3[n]$.

Semi-tight frame p-filter banks:

If the condition Eq. (17.47) is not fulfilled then the sequence $t[n]_1$ can be factorized as $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$, where $T[n]_1 \neq \tilde{T}[n]_1$. Thus, we obtain the PR p-filter bank, whose FRs are

$$\begin{aligned} &\hat{h}^0[n] = \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_0^0[n]_1, && \hat{\tilde{h}}^1[n] = -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1, \\ &\hat{h}^2[n] = T^2[n]_1, && \hat{\tilde{h}}^2[n] = \tilde{T}^2[n]_1, \\ &\hat{h}^3[n] = \omega^{-n}T^3[n]_1, && \hat{\tilde{h}}^3[n] = \omega^{-n}\tilde{T}^3[n]_1. \end{aligned} \tag{17.49}$$

where $T^2[n]_1\,\tilde{T}^2[-n]_1 = T^3[n]_1\,\tilde{T}^3[-n]_1 = t[n]_1$. The PR p-filter bank defined by Eq. (17.49) generates a semi-tight frame in the space $\Pi[N]$.

Remark 17.3 *The rational function $t[n]_1$ of ω^n is symmetric about change $n \to -n$. Therefore, it can be factorized into product of two symmetric rational functions $T[n]_1$ and $\tilde{T}[-n]_1$. An additional advantage of the semi-tight design is the option to swap approximation properties between the analysis and the synthesis framelets.*

17.3.1.2 Symmetric factorizAtion of the Matrix Q[*n*]

If the inequality in Eq. (17.47) holds then the sequence $t[n]_1$ can be factorized as $t[n]_1 = T[n]_1\,T[-n]_1$ and the following factorization of the matrix $\mathbf{Q}[n]$ is possible. Define,

$$\mathbf{P}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} A[-n]_1 & -A[-n]_1 \\ A[n]_1 & A[n]_1 \end{pmatrix} \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \mathbf{P}^{23}[n]^T, \tag{17.50}$$

where $A[n]_1 \stackrel{\text{def}}{=} T[n]_1/\sqrt{2}$. Then, the product

$$\mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \begin{pmatrix} A[-n]_1 & -A[-n]_1 \\ A[n]_1 & A[n]_1 \end{pmatrix} \cdot \begin{pmatrix} A[n]_1 & A[-n]_1 \\ -A[n]_1 & A[-n]_1 \end{pmatrix} = \mathbf{Q}[n].$$

A consequence of this choice is the fact that the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ have a linear phase:

$$\begin{aligned} &\hat{h}^2[n] = A[-n]_1 + \omega^{-n}A[n]_1, \quad \hat{h}^2[-n] = A[n]_1 + \omega^n\,A[-n]_1 = \omega^n\,\hat{h}^2[n], \\ &\hat{h}^3[n] = -A[-n]_1 + \omega^{-n}\,A[n]_1, \quad \hat{h}^3[-n] = -A[n]_1 + \omega^n\,A[-n]_1 = -\omega^n\,\hat{h}^3[n]. \end{aligned}$$

The following symmetry properties hold.

Proposition 17.1 *The impulse response (IR) of the p-filter $\mathbf{h}^2$ is symmetric about 1/2, while the IR of $\mathbf{h}^3$ is antisymmetric about 1/2.*

Proof: The IR of the p-filter $\mathbf{h}^2$ is $h^2[k] = N^{-1} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[n]$. On the other hand,

$$h^2[-k+1] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{-kn} \omega^n \hat{h}^2[n] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[-n] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[n] = h^2[k].$$

The claim about the p-filter $\mathbf{h}^3$ is proved similarly. ■

Since $\tilde{\mathbf{P}}^{23}[n] = \mathbf{P}^{23}[n]^T$, the four-channel p-filter bank

$$\begin{aligned} \hat{h}^0[n] &= \hat{h}_0^0[n]_1 + \omega^{-n} \hat{h}_1^0[n]_1, \quad \hat{h}^1[n] = -\hat{h}_1^0[-n]_1 + \omega^{-n} \hat{h}_0^0[-n]_1, \\ \hat{h}^2[n] &= A[-n]_1 + \omega^{-n} A[n]_1, \quad \hat{h}^3[n] = -A[-n]_1 + \omega^{-n} A[n]_1. \end{aligned}$$

generates a tight frame in the space $\Pi[N]$.

A symmetric factorization scheme was presented in [10]. Certainly, other schemes are possible.

17.3.1.3 Interpolating p-Filter Banks

Assume that the low-pass p-filter $\mathbf{h}^0$ is interpolating and its frequency response is $\hat{h}^0[n] = \left(1 + \omega^{-n} f[n]_1\right)/\sqrt{2}$, where the sequence $f[n]_1$ is a rational function of $\omega^n = e^{2\pi i n/N}$ that has no poles as $n \in \mathbb{Z}$ and

$$f[0]_1 = 1, \quad \omega^{-n} f[n]_1 = \omega^n \, f[-n]_1.$$

The sequence $t[n]_1$ to be factorized is

$$\begin{aligned} t[n]_1 &= 1 - \left|\hat{h}_0^0[n]_1\right|^2 + \left|\hat{h}_1^0[n]_1\right|^2 = \frac{W[n]_1}{2}, \\ W[n]_1 &\stackrel{\text{def}}{=} 1 - |f[n]_1|^2 = 2\hat{h}^0[n]\,\hat{h}^0[n + N/2]. \end{aligned}$$

When the diagonal factorization of the matrix $\mathbf{Q}[n]$ is applied, the polyphase submatrices are

$$\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & f[n]_1 \\ -f[-n]_1 & 1 \end{pmatrix}, \tag{17.51}$$

$$\mathbf{P}^{23}[n] = \begin{pmatrix} T^2[n]_1 & 0 \\ 0 & T^3[n]_1 \end{pmatrix} \quad \tilde{\mathbf{P}}^{23}[n] = \begin{pmatrix} \tilde{T}^2[n]_1 & 0 \\ 0 & \tilde{T}^3[n]_1 \end{pmatrix}, \tag{17.52}$$

where $T^2[n]_1\, \tilde{T}^2[-n]_1 = T^3[n]_1\, \tilde{T}^3[-n]_1 = W[n]_1/2$.

In the case when the inequality Eq. (17.47) holds and the sequence $t[n]_1$ is factorized as $t[n]_1 = T[n]_1\, T[-n]_1$, the polyphase matrices $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$ are given in Eq. (17.51), while the matrices $\mathbf{P}^{23}[n] = \tilde{\mathbf{P}}^{23}[n]^T$ are given in Eq. (17.50).

17.3.2 Low-Pass p-Filters

Like in the three-channel case, the key point in the design of a four-channel PR p-filter bank generating a (semi-)tight frame in the signal space $\Pi[N]$ is definition of a relevant low-pass p-filter $\mathbf{h}^0 = \tilde{\mathbf{h}}^0$. Once it is done, the rest of the p-filters is designed in a way described in Sect. 17.3.1.

17.3.2.1 Interpolating Low-Pass p-Filters

A number of interpolating linear phase low-pass p-filters, which originate from polynomial and discrete interpolating splines, is described in Sect. 17.2.3. The structure of the frequency responses of those p-filters is

$$\hat{h}^0[n] = \hat{\tilde{h}}^0[n] = \frac{1}{\sqrt{2}}\left(1 + \omega^{-n}\hat{f}[n]_1\right),$$

where $\hat{f}[n]_1$ is the frequency response of a prediction p-filter derived from a spline.

If the polynomial interpolating spline of order p is utilized then

$$\hat{f}[n]_1 = \hat{f}_c^p[n]_1 \stackrel{\text{def}}{=} \frac{v^p[n]_1}{u^p[n]_1},$$

$$v^p[n]_1 \stackrel{\text{def}}{=} \sum_{k=0}^{N/2-1} \omega^{2nk} B^p(l+1/2), \quad u^p[n]_1 \stackrel{\text{def}}{=} \sum_{k=0}^{N/2-1} \omega^{2nk} B^p(l),$$

where $B^p(t)$ is the B-spline. The corresponding low-pass p-filter $\mathbf{h}^0$ locally restores polynomials of degree $p-1$ if p is even and of degree p if p is odd.

The frequency response of the prediction p-filter derived from the discrete spline of order $2r$ is

$$\hat{f}[n]_1 = \hat{f}_d^{2r}[n]_1 \stackrel{\text{def}}{=} \frac{\omega^n\left(\cos^{2r}\pi n/N - \sin^{2r}\pi n/N\right)}{\cos^{2r}\pi n/N + \sin^{2r}\pi n/N}.$$

The corresponding low-pass p-filter $\mathbf{h}^0$ locally restores polynomials of degree $2r-1$.

All the above low-pass p-filters, except for p-filters derived from splines of second order, have infinite impulse response. These second order p-filters locally restore linear polynomials. However, in some applications, it is preferable to have FIR filters with a higher approximation accuracy. One of such p-filters is presented in Sect. 17.2.3. That is the interpolating low-pass p-filter whose FR is $\hat{h}^0[n] = \left(1 + \omega^{-n} f_q^3[n]_1\right)/\sqrt{2}$, where the prediction p-filter is derived from the quadratic quasi-interpolating spline (Eq. (16.20)):

$$f_q^3[n]_1 = \frac{-\omega^{-2n} + 9 + 9\omega^{2n} - \omega^{4n}}{16}, \tag{17.53}$$

$$\hat{h}^0[n] = \sqrt{2}\,\cos^4\frac{\pi n}{N}\left(1 + 2\sin^2\frac{\pi n}{N}\right). \tag{17.54}$$

17.3.2.2 Two Examples of Non-interpolating p-FIR p-Filters

Quadratic quasi-interpolating spline

Denote by $S^3(t)$ the quadratic spline, which quasi-interpolates the even polyphase component $\mathbf{x}_0$ of a signal $\mathbf{x}_0$ (Sect. 16.2.5.5). Denote values of the spline $s_0[k] \stackrel{\text{def}}{=} S^3(k)$ and $s_1[k] \stackrel{\text{def}}{=} S^3(k+1/2), \kappa \in \mathbb{Z}$. Then, the DFT $\hat{s}_0[n]_1 = g_q^3[n]_1\,\hat{x}_0[n]_1$ and $\hat{s}_1[n]_1 = f_q^3[n]_1\,\hat{x}_0[n]_1$, where $f_q^3[n]_1$ is given in Eq. (17.54) and

$$g_q^3[n]_1 = \frac{-\omega^{-4n} + 4\omega^{-2n} + 58 + 4\omega^{2n} - \omega^{4n}}{64}.$$

Non-interpolating low-pass p-filter is defined by its FR:

$$\begin{aligned}\hat{h}^0[n] &\stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}\left(\hat{s}_0[n]_1 + \omega^{-n}\,\hat{s}_1[n]_1\right) = \frac{1}{\sqrt{2}}\left(g_q^3[n]_1 + \omega^{-n} f_q^3[n]_1\right)\\ &= \frac{1}{\sqrt{2}}\cos^4\frac{\pi n}{N}\left(3 - \cos\frac{2\pi n}{N}\right)\\ &= \hat{h}^0[0] - \sqrt{2}\,\sin^4\frac{\pi n}{N}\left(5 + 4\cos\frac{2\pi n}{N} + \cos^2\frac{2\pi n}{N}\right).\end{aligned}$$

The IR of the p-filter $\mathbf{h}^0$ comprises nine terms (up to periodization). The low-pass p-filter $\mathbf{h}^0$ locally restores cubic polynomials.

Note that the same approximation order is achieved by p-filters, which have shorter IRs. For example, the interpolating p-filter $\mathbf{h}^0$, whose frequency response is defined in Eq. (17.54), locally restores cubic polynomials, while its IR comprises only five terms (up to periodization).

Like the interpolating spline, the quadratic quasi-interpolating spline possesses the super-convergence property at midpoints between grid nodes. This is the reason why the p-filter and $\mathbf{h}^0$ locally restores cubic polynomials, respectively.

Pseudo-spline p-filters

A family of linear phase FIR low-pass filters is introduced in [10] for non-periodic setting. In the periodic case, the design starts from the obvious identity

$$1 = \left(\cos^2 \frac{\pi n}{N} + \sin^2 \frac{\pi n}{N}\right)^{m+l}, \quad m, l \in \mathbb{N}. \tag{17.55}$$

The FR of a low-pass p-filter $\mathbf{h}^0$ is defined by summation of the first $l+1$ terms of the binomial expansion in Eq. (17.55):

$$\hat{h}^0[n] = \sqrt{2} \cos^{2m} \frac{\pi n}{N} \sum_{j=0}^{l} \binom{m+l}{j} \sin^{2j} \frac{\pi n}{N} \cos^{2(l-j)} \frac{\pi n}{N}.$$

Example: $m = 2,\ l = 1$

$$\hat{h}^0[n] = \sqrt{2} \cos^4 \frac{\pi n}{N} \left(\cos^2 \frac{\pi n}{N} + 3 \sin^2 \frac{\pi n}{N}\right) = \sqrt{2} \cos^4 \frac{\pi n}{N} \left(1 + 2 \sin^2 \frac{\pi n}{N}\right).$$

One can observe that the p-filter $\mathbf{h}^0$ is the same as the interpolating p-filter derived from the quadratic quasi-interpolating spline, whose FR is given by Eq. (17.54).

Example from [10]: $m = 3,\ l = 1$

$$\hat{h}^0[n] = \sqrt{2} \cos^6 \frac{\pi n}{N} \left(1 + 3 \sin^2 \frac{\pi n}{N}\right).$$

$$\hat{h}^0[n] - \hat{h}^0[0] = \sqrt{2} \sin^4 \frac{\pi n}{N} \left(3 \cos^4 \frac{\pi n}{N} + 2 \cos^2 \frac{\pi n}{N} + 1\right).$$

Then, Proposition 15.5 implies that the low-pass p-filter $\mathbf{h}^0$ locally restores sampled cubic polynomials. The p-filter $\mathbf{h}^0$ is not interpolating. IRs of the filters $\mathbf{h}^0$ comprises nine terms (up to periodization).

17.4 Four-Channel p-Filter Banks Using Spline Filters

In this section, we present examples of PR four-channel p-filter banks, which generate tight and semi-tight frames in the space $\Pi[N]$ of N-periodic discrete-time signals.

The design scheme was described in Sect. 17.3.1.

Summary of the design scheme

Once a linear phase low-pass filter $\mathbf{h}^0$ whose FR is $\hat{h}^0[n] = \hat{h}_0^0[n]_1 + \omega^{-n} \hat{h}_1^0[n]_1$, is designed, the high-pass filter $\mathbf{h}^1$ is defined by its FR as follows:

$$\hat{h}^1[n] = \omega^{-n}\left(\hat{h}_0^0[n]_1 - \omega^{-n}\hat{h}_1^0[n]_1\right) = \omega^{-n}\hat{h}^0[n+N/2].$$

Design of the band-pass p-filters is based on factorization of the sequence

$$t[n]_1 \stackrel{\text{def}}{=} 1 - \left|\hat{h}_0^0[n]_1\right|^2 + \left|\hat{h}_1^0[n]_1\right|^2 = T[n]_1\,\tilde{T}[-n]_1.$$

Once the factorization is accomplished, the FRs of the band-pass p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined as

$$\begin{aligned} \hat{h}^2[n] &= T[n]_1, \quad \hat{h}^3[n] = -\omega^{-n}\tilde{T}[-n]_1, \\ \hat{\tilde{h}}^2[n] &= \tilde{T}[n]_1, \quad \hat{\tilde{h}}^3[n] = -\omega^{-n}T[-n]_1. \end{aligned} \tag{17.56}$$

The PR pair $\left\{\tilde{\mathbf{H}}, \mathbf{H}\right\}$ of the p-filter banks, where $\tilde{\mathbf{H}} = \left\{\mathbf{h}^0, \mathbf{h}^1, \tilde{\mathbf{h}}^2, \tilde{\mathbf{h}}^3\right\}$ and $\tilde{\mathbf{H}} = \left\{\mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2, \mathbf{h}^3\right\}$ generates a semi-tight frame in $\Pi[N]$.

When $T[n]_1 = \tilde{T}[n]_1$, we use the *"Symmetric"* design of the p-filter banks $\mathbf{H}$ generating tight frames:

$$\hat{h}^2[n] = A[-n]_1 + \omega^{-n}A[n]_1, \quad \hat{h}^3[n] = -A[-n]_1 + \omega^{-n}\,A[n]_1, \tag{17.57}$$

where $A[n]_1 \stackrel{\text{def}}{=} T[n]_1/\sqrt{2}$.

In the forthcoming examples, we use the "Symmetric" design of tight frames.

Recall that impulse responses of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 0, 1, 2, 3$, are the framelets $\tilde{\psi}^s_{[1]}$ and $\psi^s_{[1]}$ of the first level, respectively.

In almost all the forthcoming examples, the low-pass p-filters are interpolating. If the low-pass p-filter is interpolating then the polyphase submatrices of the four-channel PR p-filter bank are presented in Eqs. (17.51), (17.52) and (17.50). In this case, once the prediction p-filter $\mathbf{f}$ is available, the FRs of the low- and the high-pass p-filters and the sequence to be factorized are

$$\begin{gathered} \hat{h}^0[n] = \hat{\tilde{h}}^0[n] = \left(1 + \omega^{-n}f[n]_1\right)/\sqrt{2}, \\ \hat{h}^1[n] = \hat{\tilde{h}}^1[n] = \omega^{-n}\,\left(1 - \omega^{n}f[-n]_1\right)/\sqrt{2}, \\ t[n]_1 = W[n]_1/2, \quad W[n]_1 \stackrel{\text{def}}{=} 1 - |f[n]_1|^2 = 2\hat{h}^0[n]\,\hat{h}^0[n+N/2]. \end{gathered}$$

Consequently, if $W[n]_1 = w[n]_1\,\tilde{w}[-n]_1$ then the sequences

$$\tilde{T}[n]_1 = \frac{\tilde{w}[n]_1}{\sqrt{2}}, \quad T[n]_1 = \frac{w[n]_1}{\sqrt{2}}. \tag{17.58}$$

If the sequence $W[n]_1$ is factorized as $W[n]_1 = w[n]_1\,w[-n]_1$ then the sequence $A[n]_1 = w[n]_1/2$.

17.4.1 Examples of p-Filter Banks With p-FIR p-Filters

A few examples of p-filter banks with FIR p-filters are given in this section.

Linear interpolating spline:

The prediction p-filter is $\hat{f}_c^2[n]_1 = (1 + \omega^{2n})/2$. Thus,

$$W[n]_1 = \frac{-\omega^{2n} + 2 - \omega^{-2n}}{4} \Longrightarrow A[n]_1 = \frac{1 - \omega^{2n}}{4}.$$

The frequency responses of the p-filters are

$$\hat{h}^0[n] = \sqrt{2}\,\cos^2\frac{\pi n}{N}, \quad \hat{h}^1[n] = \omega^{-n}\sqrt{2}\,\sin^2\frac{\pi n}{N},$$
$$\hat{h}^2[n] = \frac{1 - \omega^{-2n}}{4} + \omega^{-n}\frac{(1 - \omega^{2n}}{4} = -\frac{(1 + \omega^n)(1 - \omega^n)^2}{4\,\omega^{2n}},$$
$$\hat{h}^3[n] = -\frac{1 - \omega^{-2n}}{4} + \omega^{-n}\frac{1 - \omega^{2n}}{4} = \frac{(1 - \omega^n)(1 + \omega^n)^2}{4\,\omega^{2n}}.$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of the first degree while the p-filters $\mathbf{h}^1$ and $\mathbf{h}^2$ locally eliminate them. The p-filter $\mathbf{h}^3$ locally eliminates only constants. The framelets $\psi^1_{[1]}$ and $\psi^2_{[1]}$, which are the IRs of the p-filters $\mathbf{h}^1$ and $\mathbf{h}^2$, respectively, have two LDVMs. Either of them is symmetric. The framelet $\psi^3_{[1]}$ is antisymmetric and has one LDVM.

IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their magnitude responses are displayed in Fig. 17.9. Although both magnitude responses $|\hat{h}^2[n]|$ and $|\hat{\mathbf{h}}^3[n]|$ are equal to zero when $n = 0$, the magnitude response $|\hat{h}^2[n]|$ is flatter in the zero vicinity in comparison to $|\hat{h}^3[n]|$. This geometrical phenomenon reflects the fact that ψ^2 has two LDVMs in comparison to one LDVM in ψ^3.

Quadratic quasi-interpolating spline (interpolating low-pass p-filter):

The FRs of the prediction p-filter $\mathbf{f}$ and of low-pass p-filter $\mathbf{h}^0$ are given in Eqs. (17.53) and (17.54), respectively. Then, we have

$$\hat{h}^0[n] = \sqrt{2}\,\cos^4\frac{\pi n}{N}\left(1 + 2\sin^2\frac{\pi n}{N}\right),$$
$$\hat{h}^1[n] = \omega^{-n}\,\hat{h}^0[n + N/2] = \omega^{-n}\sqrt{2}\,\sin^4\frac{\pi n}{N}\left(1 + 2\cos^2\frac{\pi n}{N}\right).$$

The sequence $W^1[n]$ to be factorized is

$$W^1[n] = \left(\frac{\omega^{2n} - 2 + \omega^{-2n}}{16}\right)^2 \Gamma_c^4[n]_1,$$
$$\Gamma_c^4[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 14 - \omega^{-2n}. \tag{17.59}$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of third degree while the p-filter $\mathbf{h}^1$ locally eliminates them. Thus, the framelets $\psi^1_{[1]}$ has four LDVMs.

The following factorization modes $W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$ are possible:

1. *Non-symmetric factorization* $W[n]_1 = w[n]_1\, w[-n]_1$, where

$$w[n]_1 = \frac{(\omega^{-2n} - 2 + \omega^{2n})\,(1 - q\,\omega^{2n})}{16\sqrt{q}}, \quad q = 7 - 4\sqrt{3}.$$

   FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) with $A[n]_1 = w[n]_1/2$.

   The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\},\ s = 0, 1, 2, 3$, generates a tight frame. The p-filters $\mathbf{h}^3$ and $\mathbf{h}^2$ locally eliminate sampled polynomials of second and first degrees, respectively. The framelets $\psi^3_{[1]}$ and $\psi^2_{[1]}$, which are impulse responses of the p-filters $\mathbf{h}^1$, $\mathbf{h}^3$ and $\mathbf{h}^2$, respectively, have three and two LDVMs, respectively.

   IRs of the p-filters $\mathbf{h}^s,\ s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.9 (second from top). MRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ mirror each other.
2. *Symmetric factorization* of $W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$, which provides equal number (two) of LDVMs to the analysis and to the synthesis framelets:

$$\tilde{w}[n]_1 = \frac{\left(\omega^{-2n} - 2 + \omega^{2n}\right)}{4}, \quad w[n]_1 = \frac{\left(\omega^{-2n} - 2 + \omega^{2n}\right)\ \Gamma_c^4[n]_1}{64}.$$

   FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s,\ s = 2, 3$, are given in Eq. (17.56), where the sequences $T[n]_1$ and $\tilde{T}[n]_1$ are defined by Eq. (17.58).
3. *Antisymmetric factorization* of $W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$, which assigns three LDVMs to the analysis framelet $\tilde{\psi}^2_{[1]}$ leaving only one LDVM to the synthesis framelet $\psi^2_{[1]}$ and vice versa for the framelets $\tilde{\psi}^3_{[1]}$ and $\psi^3_{[1]}$:

$$\tilde{w}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \quad w[n]_1 = \frac{\left(1 - \omega^{2n}\right)\ \Gamma_c^4[n]_1}{32}.$$

   FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s,\ s = 2, 3$, are given in Eq. (17.56), where the sequences $T[n]_1$ and $\tilde{T}[n]_1$ are defined by Eq. (17.58).

Figure 17.5 displays impulse responses of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are the discrete time framelets of the first level and their MR.

Quadratic quasi-interpolating spline (non-interpolating low-pass p-filter):

Frequency responses of the p-filter $\mathbf{h}^0$ and $\mathbf{h}^1$ are

$$\hat{h}^0[n] = \frac{1}{\sqrt{2}} \cos^4 \frac{\pi n}{N} \left(3 - \cos \frac{2\pi n}{N}\right),$$
$$\hat{h}^1[n] = \frac{\omega^{-n}}{\sqrt{2}} \sin^4 \frac{\pi n}{N} \left(3 + \cos \frac{2\pi n}{N}\right).$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of third degree while the p-filter $\mathbf{h}^1$ locally eliminates them. Thus, the framelets $\psi^1_{[1]}$ has four LDVMs.

$$t[n]_1 \stackrel{\text{def}}{=} 1 - |h^0_0[n]_1|^2 - |h^0_1[n]_1|^2 = \frac{2(\omega^{2n} - 2 + \omega^{-2n})^2\, \Upsilon[n]_1}{128^2},$$
$$\Upsilon[n]_1 \stackrel{\text{def}}{=} -\omega^{4n} - 12\omega^{2n} + 346 - 12\omega^{-2n} - \omega^{-4n}.$$

1. *Non-symmetric factorization* $t[n]_1 = T[n]_1\, T[-n]_1$:

$$T[n]_1 = \frac{\sqrt{2}\,(\omega^{2n} - 2 + \omega^{-2n})(1 - \alpha_1\omega^{2n})(1 + \alpha_2\omega^{2n})}{128\sqrt{\alpha_1\,\alpha_2}},$$
$$\alpha_1 = 0.07395375302024236412202494176 4069,$$
$$\alpha_2 = 0.03912854562754878052646969481 2049.$$

   FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) with $A[n]_1 = T[n]_1/\sqrt{2}$. The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\}$, $s = 0, 1, 2, 3$, generates a tight frame. The framelet $\psi^3_{[1]}$ has three LDVMs, while $\psi^2_{[1]}$ has two. IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.9 (center).
2. *Symmetric factorization* $t[n]_1 = T[n]_1\, \tilde{T}[-n]_1$, where either of framelets $\tilde{\psi}^s_{[1]}$ and $\psi^s_{[1]}$, $s = 2, 3$, has two LDVMs:

$$T[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n})\, \Upsilon[n]_1}{2048},$$
$$\tilde{T}[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n})}{4}.$$

3. *Antisymmetric factorization* of $t[n]_1 = T[n]_1\, \tilde{T}[-n]_1$, which assigns three LDVMs to the analysis framelet $\tilde{\psi}^2_{[1]}$ leaving only one LDVM to the synthesis framelet $\psi^2_{[1]}$ and vice versa for the framelets $\tilde{\psi}^3_{[1]}$ and $\psi^3_{[1]}$:

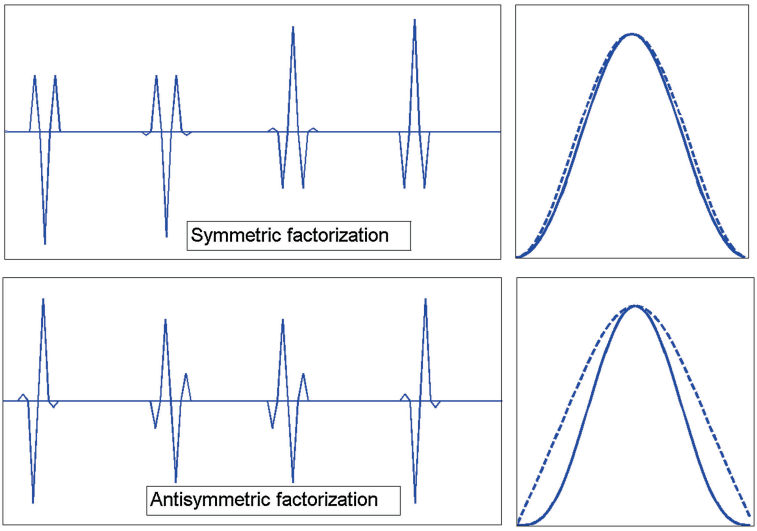


Fig. 17.5 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \mathbf{h}^3 \longrightarrow \tilde{\mathbf{h}}^3$ derived from quadratic quasi-interpolating splines (interpolating low-pass p-filter). *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

$$\tilde{T}[n]_1 = \sqrt{2}\,\frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8},$$
$$T[n]_1 = \sqrt{2}\,\frac{\left(1 - \omega^{2n}\right)\,\Upsilon[n]_1}{1024}.$$

FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined in Eq. (17.56). Figure 17.6 displays IRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are discrete-time framelets of the first level, and their MRs.

Upgraded quadratic quasi-interpolating spline:

Frequency response of the prediction p-filter $\mathbf{f} = \mathbf{f}_u^3$ is given in Eq. (16.21).

Then, frequency responses of the low- and high-pass p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$, respectively, are

$$\hat{h}^0[n] = \frac{1 + \omega^{-n}\, f_u^3[n]_1}{\sqrt{2}} = \sqrt{2}\cos^6\frac{\pi n}{N}\left(1 + 9\sin^2\frac{\pi n}{N} - \frac{3}{2}\sin^2\frac{2\pi n}{N}\right),$$
$$\hat{h}^1[n] = \omega^{-n}\,\hat{h}^0[n + N/2] = \omega^{-n}\,\sqrt{2}\sin^6\frac{\pi n}{N}\left(1 + 9\cos^2\frac{\pi n}{N} - \frac{3}{2}\sin^2\frac{2\pi n}{N}\right).$$

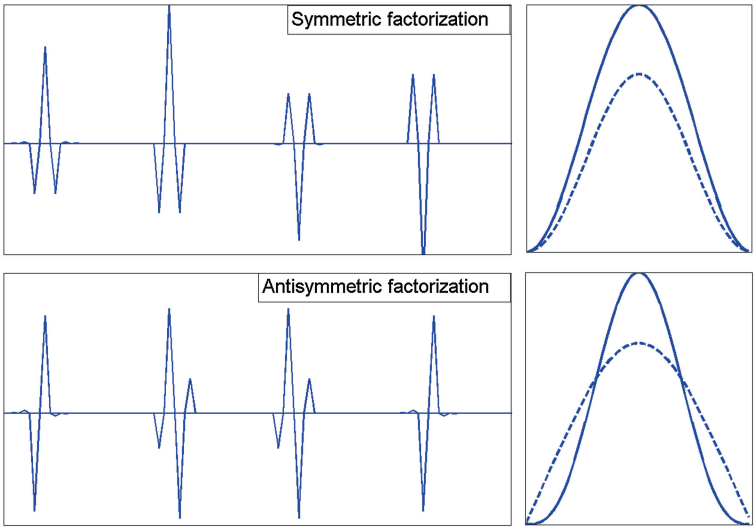


Fig. 17.6 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \mathbf{h}^3 \longrightarrow \tilde{\mathbf{h}}^3$ derived from quadratic quasi-interpolating splines (non-interpolating low-pass p-filter). *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

Their IRs comprise seven terms (up to periodization). The low-pass $\mathbf{h}^0$ and the high-pass $\mathbf{h}^1$ p-filters locally restore and eliminate sampled polynomials of fifth degree, respectively. Thus the framelet $\psi^1_{[1]}$ has six LDVMs.

The sequence

$$W[n]_1 = \frac{(-\omega^{2n} + 2 - \omega^{-2n})^3}{256^2}\, P[n]_1, \quad P[n]_1 \stackrel{\text{def}}{=} 9\omega^{4n} - 96\omega^{2n} + 814 - 96\omega^{-2n} + 9\omega^{-4n}.$$

1. *Non-symmetric factorization* $W[n]_1 = w[n]_1\, \omega[-n]_1$: The Laurent polynomial of ω^{2n}, $P[n]_1$ has four roots:

$$\alpha_1 = \sqrt{15}i + \frac{4}{3}\sqrt{2\sqrt{15}\,i - 5} + \frac{8}{3} \simeq 5.274196044905089 + 7.833803759614890i,$$
$$\alpha_2 = 1/\alpha_1, \quad \alpha_3 = \alpha_1^*, \quad \alpha_4 = 1/\alpha_1^*,$$

where a^* is the complex conjugate of a. Thus, the sequence $W[n]_1$ can be factorized as follows:

$$W[n]_1 = w[n]_1\, w[-n]_1, \quad w[n]_1 = \frac{3(1-\omega^{-2n})^3}{5256}\left(\beta_0 - \beta_1\,\omega^{2n} + \beta_2\,\omega^{4n}\right),$$
$$\beta_0 = \frac{1}{|\alpha_1|} \simeq 0.105889419948135, \quad \beta_1 = \frac{\alpha_1 + \alpha_1^*}{|\alpha_1|} \simeq 1.116963119775494,$$
$$\beta_2 = |\alpha_1| \simeq 9.443814126943037.$$

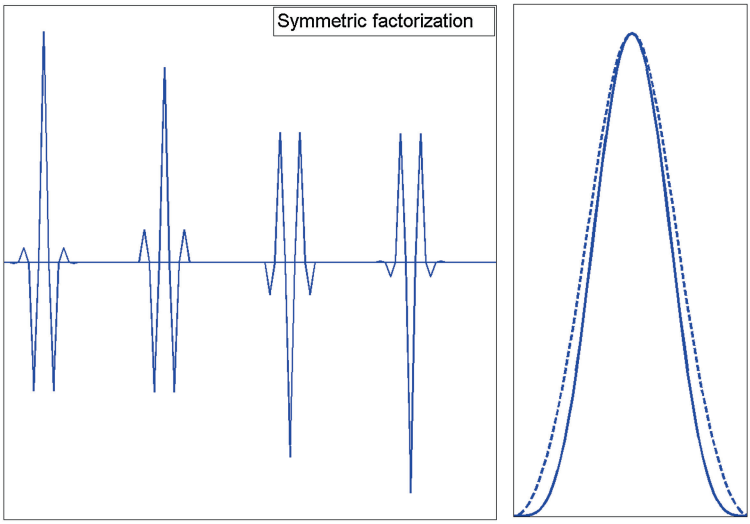


Fig. 17.7 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \tilde{\mathbf{h}}^3 \longrightarrow \mathbf{h}^3$ derived from the upgraded quadratic quasi-interpolating splines. *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) with $A[n]_1 = w[n]_1/2$.

The p-filter $\mathbf{h^3}$ eliminates sampled polynomials of second degree, thus the framelet $\psi^3_{[1]}$ has three LDVMs. The p-filter $\mathbf{h^2}$ eliminates cubic polynomials, thus the framelet $\psi^2_{[1]}$ has four LDVMs.

IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their magnitude responses are displayed in Fig. 17.9 (second from bottom).

2. *Symmetric factorization* $W[n]_1 = w[n]_1\, \tilde{\omega}[-n]_1$, which assigns four LDVMs to the analysis framelet $\tilde{\psi}^2_{[1]}$ and two LDVMs to the synthesis framelet $\psi^2_{[1]}$ and vice versa for the framelets $\tilde{\psi}^3_{[1]}$ and $\psi^3_{[1]}$:

$$\tilde{w}[n]_1 = \frac{(-\omega^{2n} + 2 - \omega^{-2n})^2}{256} 4, \quad w[n]_1 = \frac{(-\omega^{2n} + 2 - \omega^{-2n})}{256}\, P[n]_1.$$

FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are given in Eq. (17.56), where the sequences $T[n]_1$ and $\tilde{T}[n]_1$ are defined by Eq. (17.58).

Figure 17.7 displays IRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are discrete-time framelets of the first level and their MRs.

Pseudo-spline:

FRs of the non-interpolating low-pass p-filter $\mathbf{h}^0$ and the corresponding high-pass p-filter $\mathbf{h}^1$ are

$$\hat{h}^0[n] = \sqrt{2}\cos^6\frac{\pi n}{N}\left(1+3\sin^2\frac{\pi n}{N}\right),$$
$$\hat{h}^1[n] = \omega^{-n}\sqrt{2}\sin^6\frac{\pi n}{N}\left(1+3\cos^2\frac{\pi n}{N}\right),$$
$$.$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of third degree while the p-filter $\mathbf{h}^1$ locally eliminates sampled polynomials of fifth degree. Thus, the framelets $\psi^1_{[1]}$ has six LDVMs. We have

$$t[n]_1 \stackrel{\text{def}}{=} 1-|h^0_0[n]_1|^2-|h^0_1[n]_1|^2 = \frac{2(\omega^{2n}-2+\omega^{-2n})^2\,Q[n]_1}{256^2},$$
$$Q[n]_1 \stackrel{\text{def}}{=} -9\omega^{4n}-28\omega^{2n}+1610-28\omega^{-2n}-9\omega^{-4n}.$$

1. *Non-symmetric factorization* $t[n]_1 = T[n]_1\,T[-n]_1$:

$$T[n]_1 = \frac{\sqrt{2}\,3(\omega^{2n}-2+\omega^{-2n})(1-\alpha_1\omega^{2n})(1+\alpha_2\omega^{2n})}{256\sqrt{\alpha_1\,\alpha_2}}$$
$$\alpha_1 = 0.08403672131163586375139019744678 5,$$
$$\alpha_2 = 0.06654171895289296120728701185405 9.$$

FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) with $A[n]_1 = T[n]_1/\sqrt{2}$. The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\},\ s = 0, 1, 2, 3$, generates a tight frame. The p-filters $\mathbf{h^2}$ and $\mathbf{h^3}$ locally eliminate sampled polynomials of first and second degrees, respectively, thus the framelets $\psi^2_{[1]}$ and $\psi^3_{[1]}$ have two and three LDVMs, respectively.

IRs of the p-filters $\mathbf{h}^s,\ s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.9 (bottom).

2. *Symmetric factorization* $t[n]_1 = T[n]_1\,\tilde{T}[-n]_1$:

$$T[n]_1 = \frac{\sqrt{2}(-\omega^{2n}+2-\omega^{-2n})\,Q[n]_1}{8192},$$
$$\tilde{T}[n]_1 = \frac{\sqrt{2}(-\omega^{2n}+2-\omega^{-2n})}{4}.$$

The p-filters $\mathbf{h^s}$ and $\tilde{\mathbf{h}}^s, : s = 2, 3$, locally eliminate sampled polynomials of first degree, thus the framelets $\psi^2_{[1]}$ and $\psi^s_{[1]}$ have two LDVMs.

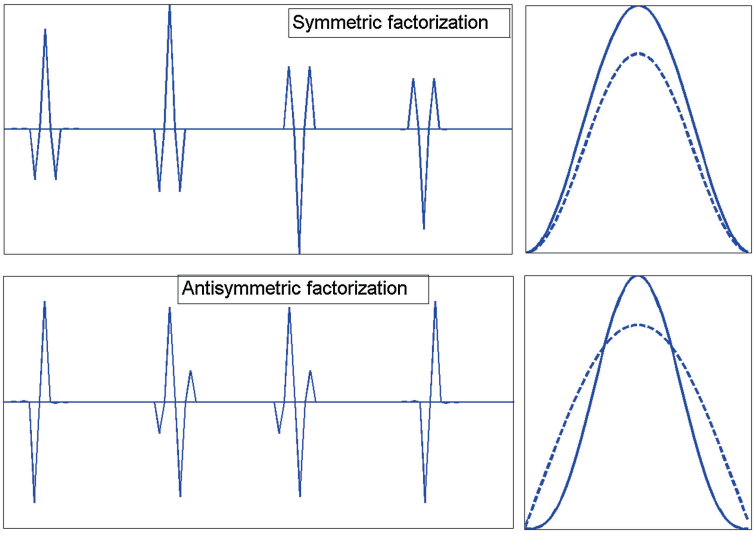


Fig. 17.8 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \mathbf{h}^3 \longrightarrow \tilde{\mathbf{h}}^3$ derived from the pseudo-spline. *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

3. Antisymmetric factorization of $t[n]_1 = T[n]_1\,\tilde{T}[-n]_1$, which assigns three LDVMs to the analysis framelet $\tilde{\psi}^2_{[1]}$ leaving only one LDVM to the synthesis framelet $\psi^2_{[1]}$ and vice versa for the framelets $\tilde{\psi}^3_{[1]}$ and $\psi^3_{[1]}$:

$$\tilde{T}[n]_1 = \sqrt{2}\,\frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \quad T[n]_1 = \sqrt{2}\,\frac{\left(1-\omega^{2n}\right)\,Q[n]_1}{4096}.$$

FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined in Eq. (17.56).

Figure 17.8 displays the IRs of the p-filters $\mathbf{h}^s$, which are discrete-time framelets of the first level, and their MR.

Display of tight frame FIR p-filters

Figure 17.9 displays impulse and magnitude responses of the p-FIR four-channel p-filter banks that generate tight frames. The impulse responses of the p-filters $\mathbf{h}^s$ are the corresponding discrete-time framelets $\psi^s_{[1]}[k]$, $s = 0, 1, 2, 3$, of the first decomposition level. The framelets $\psi^s_{[1]}[k]$, $s = 0, 1, 2$, are symmetric, while the framelets $\psi^3_{[1]}[k]$ are antisymmetric. The magnitude responses of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ mirror each other and the same is true for the band-pass pair $\mathbf{h}^2$ and $\mathbf{h}^3$. In the caption to the figure, abbreviation QIS means quasi-interpolating spline.

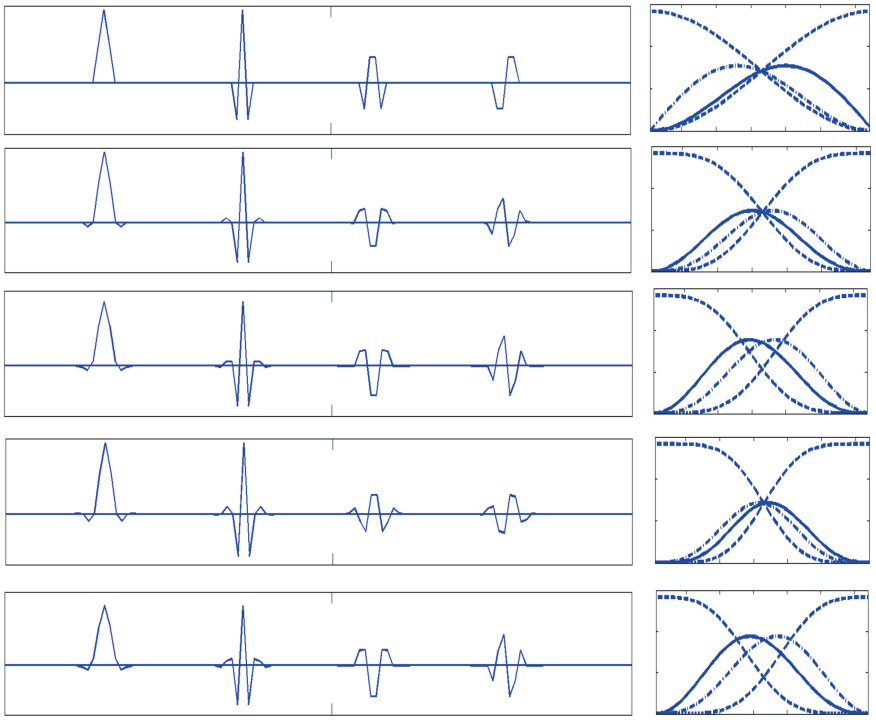


Fig. 17.9 Impulse and magnitude responses of the p-FIR four-channel p-filter banks that generate tight frames. Left pictures display the impulse responses of p-filters where, from *left* to *right*, we have: $\mathbf{h}^0 \longrightarrow \mathbf{h}^1 \longrightarrow \mathbf{h}^2 \longrightarrow \mathbf{h}^3$. *Right* pictures display the magnitude responses of these p-filters: $\mathbf{h}^0$ and $\mathbf{h}^2$ (*dashed lines*), $\mathbf{h}^3$ (*dash-dot line*) and $\mathbf{h}^2$ (*solid line*). *Top* linear spline. Second from *top* quadratic QIS (interpolating low-pass p-filter). *Center* quadratic QIS (non-interpolating low-pass p-filter). Second from the *bottom*: upgraded quadratic QIS. *Bottom* pseudo-spline

17.4.2 Four-Channel p-Filter Banks With IIR p-Filters

Unlike non-periodic setting, implementation cost of IIR p-filters is no higher than the cost of p-FIR p-filters. However, giving up the requirement of finite impulse response provides additional flexibility in the design of p-filter banks with necessary properties.

For the design, we use the interpolating low-pass p-filters $\mathbf{h}^0$. The frequency responses of the p-filters are structured as $\hat{h}^0[n] = \left(1 + \omega^{-n}\hat{f}[n]_1\right)/\sqrt{2}$, where $\hat{f}[n]_1$ is the frequency response of a some prediction p-filter. Prediction p-filters derived from interpolating polynomial and discrete splines are described in Sect. 16.2 and 16.3, respectively.

Similarly to design of three-channel p-filter banks with interpolating low-pass p-filters, which is described in Sect. 17.2, the key point is factorization of the sequence $W[n]_1 = 1 - |\hat{f}[n]_1|^2$. Therefore, we can use the factorization results from Sect. 17.2.

17.4.2.1 Examples of p-Filter Banks Originating From Polynomial Interpolating Splines

Quadratic interpolating spline $p = 3$:

Denote $\Omega_c^3[n]_1 \stackrel{\text{def}}{=} \omega^{2n} + 6 + \omega^{-2n}$. The prediction p-filter $\mathbf{f}_c^3$ is defined in Eq. (17.21). Frequency responses of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ and the sequence $W[n]_1$ are

$$\hat{h}^0[n] = \sqrt{2}\,\frac{\cos^4 \pi n/N}{\Omega_c^3[n]_1}, \quad \hat{h}^1[n] = \omega^{-n}\,\sqrt{2}\,\frac{\sin^4 \pi n/N}{\Omega_c^3[n]_1},$$
$$W[n]_1 = \left(\frac{\omega^{2n} - 2 + \omega^{-2n}}{\Omega_c^3[n]_1}\right)^2 = \left(\frac{\sin^2 2\pi n/N}{2\Omega_c^3[n]_1}\right)^2.$$

The p-filter $\mathbf{h}^0$ locally restores sampled cubic polynomials while the p-filter $\mathbf{h}^1$ locally eliminates them. Thus, the framelet $\psi_{[1]}^1$ has four LDVMs.

1. *Symmetric factorization* $W[n]_1 = w[n]_1\, w[-n]_1$, where

$$w[n]_1 = \frac{(\omega^{-2n} - 2 + \omega^{2n})}{\Omega_c^3[n]_1} = \frac{\sin^2 2\pi n/N}{2\Omega_c^3[n]_1}.$$

   FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) where $A[n]_1 = w[n]_1/2$. The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\}$, $s = 0, 1, 2, 3$, generates a tight frame. The p-filter $\mathbf{h}^3$ locally eliminates sampled quadratic polynomials, thus the framelet $\psi_{[1]}^3$, has three LDVMs, while the framelet $\psi_{[1]}^2$, has two LDVMs. The IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.13 (top).
2. Antisymmetric factorization of $W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$, which assigns three LDVMs to the analysis framelet $\tilde{\psi}_{[1]}^2$ leaving only one LDVM to the synthesis framelet $\psi_{[1]}^2$ and vice versa for the

$$\tilde{w}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{2\Omega_c^3[n]_1}, \quad w[n]_1 = \frac{2\left(1 - \omega^{2n}\right)}{\Omega_c^3[n]_1}.$$

   FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined in Eq. (17.56), taking into account Eq. (17.58). Figure 17.10 displays IRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are discrete-time framelets of the first level, and their MRs.

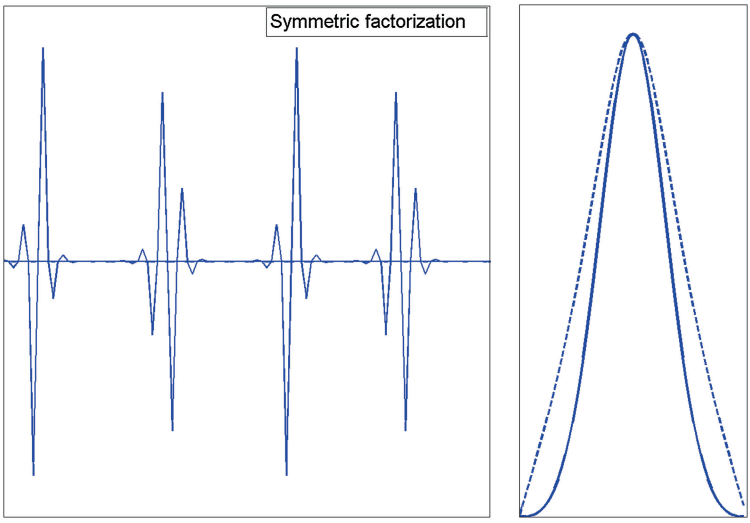


Fig. 17.10 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \tilde{\mathbf{h}}^3 \longrightarrow \mathbf{h}^3$ derived from quadratic interpolating splines. *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

Cubic interpolating spline: $p = 4$:

Denote,

$$\Omega_c^4[n]_1 \stackrel{\text{def}}{=} \omega^{2n} + 4 + \omega^{-2n}, \quad \Gamma_c^4[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 14 - \omega^{-2n}. \tag{17.60}$$

The prediction p-filter $\mathbf{f}_c^4$ is defined in Eq. (17.23). Frequency responses of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ and the sequence $W[n]_1$ are

$$\begin{aligned}
\hat{h}^0[n] &= \sqrt{2}\,\frac{\cos^4 \pi n/N\ (2 + \cos 2\pi n/N)}{\Omega_c^4[n]_1}, \\
\hat{h}^1[n] &= \omega^{-n}\,\sqrt{2}\,\frac{\sin^4 \pi n/N\ (2 - \cos 2\pi n/N)}{\Omega_c^4[n]_1}, \\
W[n]_1 &= \left(\frac{\omega^{2n} - 2 + \omega^{-2n}}{8\Omega_c^4[n]_1}\right)^2 \Gamma_c^4[n]_1.
\end{aligned} \tag{17.61}$$

The p-filter $\mathbf{h}^0$ locally restores sampled cubic polynomials while the p-filter $\mathbf{h}^1$ locally eliminates them. Thus, the framelet $\psi_{[1]}^1$ has four LDVMs.

Comparing Eq. (17.61) with Eq. (17.59), we observe that numerators of the sequences $W[n]_1$ in both cases are the same. Therefore, factorization of $W[n]_1$

for the cubic interpolating spline is similar to the factorization for the quadratic quasi-interpolating spline.

1. *Non-symmetric factorization* $W[n]_1 = w[n]_1\, w[-n]_1$, where

$$w[n]_1 = \frac{\omega^{2n} - 2 + \omega^{-2n}}{8\sqrt{q}\,\Omega_c^4[n]_1}\left(1 - q\omega^{2n}\right), \quad q = 7 - 4\sqrt{3}.$$

   FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) where $A[n]_1 = w[n]_1/2$. The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\}$, $s = 0, 1, 2, 3$, generates a tight frame. The p-filter $\mathbf{h}^3$ locally eliminates sampled quadratic polynomials, thus the framelet $\psi^3_{[1]}$, has three LDVMs, while the framelet $\psi^2_{[1]}$, has two LDVMs. IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.13 (second from top).
2. *Symmetric factorization* $W[n]_1 = w[n]_1\, \tilde{w}[-n]_1$, where

$$\tilde{w}[n]_1 = \frac{\omega^{2n} - 2 + \omega^{-2n}}{8\,\Omega_c^4[n]_1}, \quad w[n]_1 = \tilde{w}[n]_1\, \Gamma_c^4[n]_1.$$

   The p-filters $\mathbf{h^s}$ and $\tilde{\mathbf{h}}^s$, : $s = 2, 3$, locally eliminate sampled polynomials of first degree, thus the framelets $\psi^2_{[1]}$ and $\psi^s_{[1]}$ have two LDVMs.
3. *Antisymmetric factorization* $W[n]_1 = A[n]_1\, \tilde{A}[-n]_1$, which assigns three LDVMs to the analysis $\tilde{\psi^2}_{[1]}$ and one LDVM to the synthesis $\psi^2_{[1]}$ framelets and vice versa for $\tilde{\psi^3}_{[1]}$ and one LDVMs in synthesis $\psi^3_{[1]}$:

$$\tilde{A}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{4\,\Omega_c^4[n]_1}, \quad A[n]_1 = \frac{\left(1 - \omega^{2n}\right)\,\Gamma_c^4[n]_1}{16\,\Omega_c^4[n]_1}.$$

In both the symmetric and the antisymmetric cases, FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined in Eq. (17.56), taking into account Eq. (17.58). Figure 17.11 displays IRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are discrete- time framelets of the first level, and their MRs.

Interpolating spline of fourth degree, $p = 5$:

Denote,

$$\Omega_c^5[n]_1 \stackrel{\text{def}}{=} \omega^{4n} + 76\omega^{2n} + 230 + 76\omega^{-2n} + \omega^{-4n}, \quad \Gamma_c^5[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 98 - \omega^{-2n}.$$

The prediction p-filter $\mathbf{f}_c^5$ is defined in Eq. (16.16). Frequency responses of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ and the sequence $W[n]_1$ are

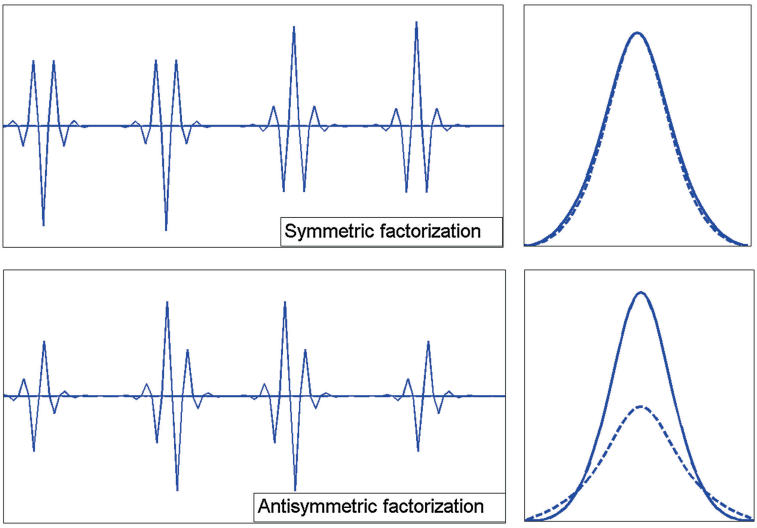


Fig. 17.11 *Left* IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \mathbf{h}^3 \longrightarrow \tilde{\mathbf{h}}^3$ derived from cubic interpolating splines. *Right* MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (*dashed lines*), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (*solid line*)

$$\hat{h}^0[n] = 128\sqrt{2}\,\frac{\cos^6 \pi n/N\,(2+\cos^2 \pi n/N)}{\Omega_c^5[n]_1},$$

$$\hat{h}^1[n] = \omega^{-n}\,128\sqrt{2}\,\frac{\sin^6 \pi n/N\,(2+\sin^2 \pi n/N)}{\Omega_c^5[n]_1},$$

$$W[n]_1 = \left(-\omega^{2n}+2-\omega^{-2n}\right)^3\,\frac{\Gamma_c^5[n]_1}{\left(\Omega_c^5[n]_1\right)^2}.$$

The p-filter $\mathbf{h}^0$ locally restores sampled polynomials of fifth degree while the p-filter $\mathbf{h}^1$ locally eliminates them. Thus, the framelet $\psi^1_{[1]}$ has six LDVMs.

1. *Non-symmetric factorization* $W[n]_1 = w[n]_1\,w[-n]_1$, where

$$w[n]_1 = \frac{\omega^{2n}-3+3\omega^{-2n}-\omega^{-4n}}{\sqrt{q}\,\Omega_c^5[n]_1}\left(1-q\omega^{2n}\right),$$

   $q = 49-20\sqrt{6}$. FRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are defined in Eq. (17.57) where $A[n]_1 = w[n]_1/2$. The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\}$, $s = 0, 1, 2, 3$, generates a tight frame. The framelets $\psi^2_{[1]}$ has four LDVMs and $\psi^3_{[1]}$ has three LDVMs. IRs of the p-filters $\mathbf{h}^s$, $s = 0, 1, 2, 3$, and their MRs are displayed in Fig. 17.13 (third from top).
2. Symmetric factorization $W[n]_1 = w[n]_1\,\tilde{w}[-n]_1$, which assigns four LDVMs to the analysis $\tilde{\psi}^2_{[1]}$ and two LDVMs to the synthesis $\psi^2_{[1]}$ framelets and vice

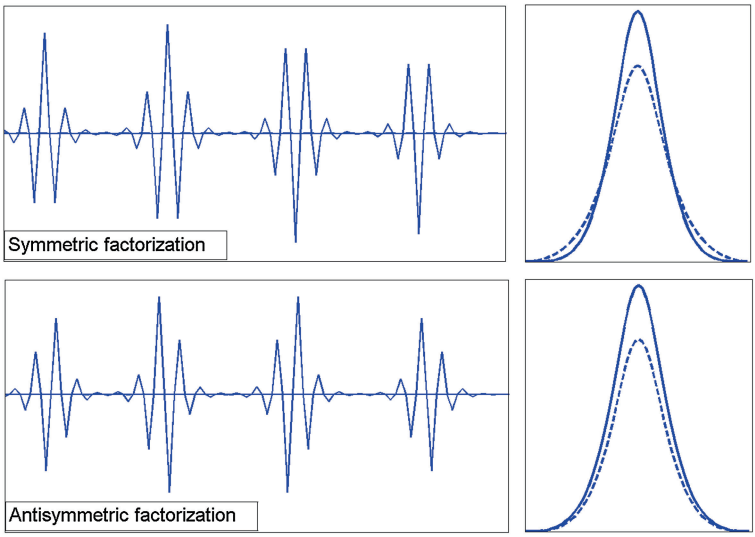


Fig. 17.12 Left: IRs of the p-filters $\mathbf{h}^2 \longrightarrow \tilde{\mathbf{h}}^2 \longrightarrow \mathbf{h}^3 \longrightarrow \tilde{\mathbf{h}}^3$ derived from interpolating splines of fifth order (fourth degree). Right: MRs of $\mathbf{h}^2$ and $\tilde{\mathbf{h}}^3$ (coincide with each other) (dashed lines), and MRs of $\tilde{\mathbf{h}}^2$ and $\mathbf{h}^3$ (coincide with each other) (solid line)

versa for $\tilde{\psi}^3_{[1]}$ and $\psi^3_{[1]}$:

$$w[n]_1 = -\frac{\omega^{2n} - 2 + \omega^{-2n}}{8\sqrt{2}\,\Omega_c^5[n]_1}\,\Gamma_c^5[n]_1, \quad \tilde{w}[n]_1 = 8\sqrt{2}\,\frac{\left(\omega^{2n} - 2 + \omega^{-2n}\right)^2}{\Omega_c^5[n]_1}.$$

3. Antisymmetric factorization $W[n]_1 = w[n]_1\,\tilde{w}[-n]_1$, where all the framelets $\tilde{\psi}^s_{[1]}$ and $\psi^s_{[1]}$, $s = 2, 3$, have three LDVMs:

$$\tilde{w}[n]_1 = \frac{\omega^{2n} - 3 + 3\omega^{-2n} - \omega^{-4n}}{8\sqrt{2}\;\Omega_c^5[n]_1}, \quad w[n]_1 = \tilde{w}[n]_1\,\frac{\Gamma_c^5[n]_1}{128}. \tag{17.62}$$

In both the symmetric and the antisymmetric cases, FRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, are defined in Eq. (17.56), taking into account Eq. (17.58). Figure 17.12 displays IRs of the p-filters $\tilde{\mathbf{h}}^s$ and $\mathbf{h}^s$, $s = 2, 3$, which are the discrete time framelets of the first level, and their MR.

17.4.2.2 Four-Channel p-Filter Banks Derived From discrete splines

Denote, $\Omega_d^{2r}[n]_1 \stackrel{\text{def}}{=} \cos^{2r} \pi n/N + \sin^{2r} \pi n/N$. The prediction p-filters $\mathbf{f} = \mathbf{f}_d^{2r}$ are given in Eq. (17.34). Frequency responses of the low- and high-pass p-filters and

the sequence $W[n]_1$ are

$$\hat{h}^0[n] = \frac{\sqrt{2}\cos^{2r}\pi n/N}{\Omega_d^{2r}[n]_1}, \quad \hat{h}^1[n] = \omega^{-n}\,\frac{\sqrt{2}\sin^{2r}\pi n/N}{\Omega_d^{2r}[n]_1}$$

$$W[n]_1 = \frac{4\,\sin^{2r} 2\pi n/N}{2^{2r}\,\left(\Omega_d^{2r}[n]_1\right)^2} = \frac{4(-1)^r\,\omega^{-2nr}\,\left(1-\omega^{2n}\right)^{2r}}{4^{2r}\,\left(\Omega_d^{2r}[n]_1\right)^2}.$$

The sequence $W[n]_1$ is factorized as $W[n]_1 = w[n]_1\, w[-n]_1$, where

$$w[n]_1 = \begin{cases} 2\,\omega^{-2np}\,\left(1-\omega^{2n}\right)^r/(4^r\,\Omega_d^{2r}[n]_1) & \text{if } r \text{ is odd, } p \in \mathbb{Z}; \\ 2\,\sin^r 2\pi n/N/(2^r\,\Omega_d^{2r}[n]_1), & \text{if } r \text{ is even,} \end{cases}$$

$$h^2[n] = A[-n]_1 + \omega^{-n}\,A[n]_1, \quad h^3[n] = -A[-n]_1 + \omega^{-n}\,A[n]_1,$$

where $A[n]_1 = w[n]_1/2$. The interpolating low-pass p-filter $\mathbf{h}^0$ locally restores sampled polynomials of degree $2r-1$ while the high-pass p-filter $\mathbf{h}^1$ eliminates them (the framelet $\psi^1_{[1]}$ has $2r$ LDVMs). The p-filter bank $\mathbf{H} = \{\mathbf{h}^s\}$, $s = 0, 1, 2, 3$, generates a tight frame. If r is even then the band-pass p-filter $\mathbf{h}^2$ locally eliminates sampled polynomials of degree r (the framelet $\psi^2_{[1]}$ has $r+1$ LDVMs), while p-filter $\mathbf{h}^3$ eliminates polynomials of degree $r-1$ (the framelet $\psi^3_{[1]}$ has r LDVMs) and vice versa for r odd.

Example: Discrete spline of fourth order: $r = 2$

In this case, the p-filter bank $\mathbf{H}$ is the same as the p-filter bank derived from the quadratic polynomial interpolating spline.

Example: Discrete spline of sixth order: $r = 3$.

In this case,

$$A[n]_1 = \frac{\omega^{-2n} - 3 + 3\omega^{2n} - \omega^{4n}}{4\left(3\,\omega^{2n} + 10 + 3\,\omega^{-2n}\right)}. \tag{17.63}$$

Example: Discrete spline of eighth order: $r = 4$.

In this case,

$$A[n]_1 = \frac{\sin^4 2\pi n/N}{16\,(\cos^8 \pi n/N + \sin^8 \pi n/N)}. \tag{17.64}$$

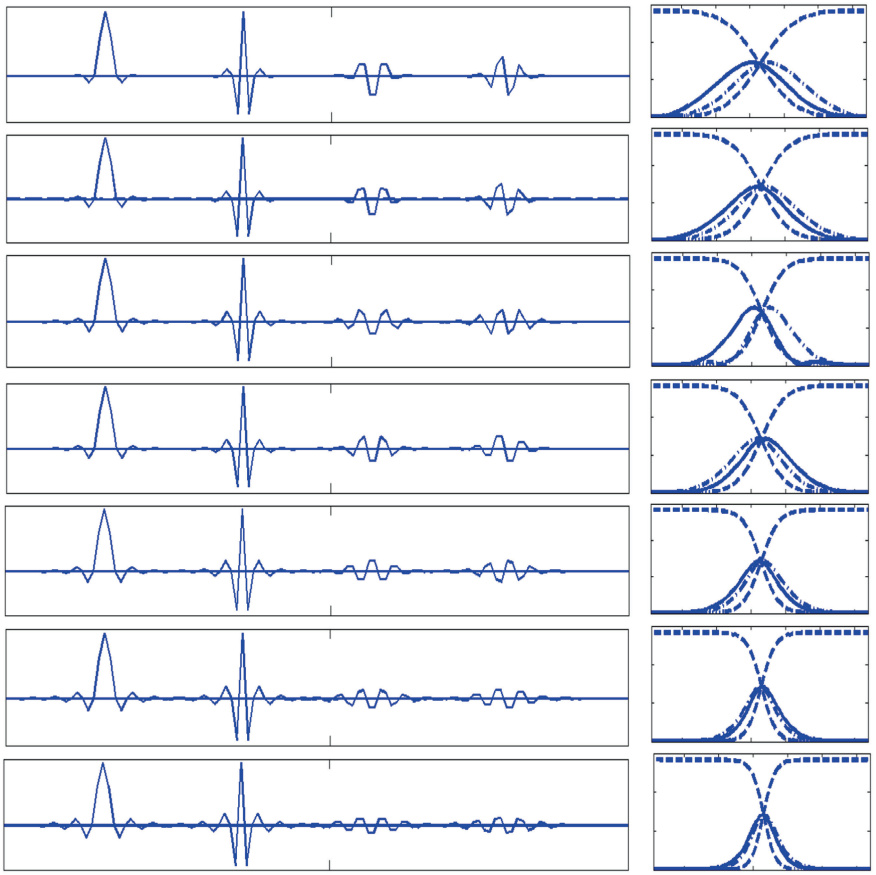


Fig. 17.13 Impulse and magnitude responses of the IIR p-filter banks that generate tight frames. *Left* pictures display impulse responses of the p-filters. *Left* to *right* in the left picture: $\mathbf{h}^0 \longrightarrow \mathbf{h}^1 \longrightarrow \mathbf{h}^2 \longrightarrow \mathbf{h}^3$. *Right* pictures display magnitude responses of these p-filters: $\mathbf{h}^0$ and $\mathbf{h}^1$ (*dashed lines*), $\mathbf{h}^3$ (*dash-dot line*) and $\mathbf{h}^2$ (*solid line*). *Top* quadratic IS. Second from *top* cubic IS. Third from *top* four degree IS. *Center* DS of order 6. Third from *bottom* DS of order 8. Second from *bottom* DS of order 10. *Bottom* discrete spline of order 12

Display of tight frame IIR p-filters

Figure 17.13 displays impulse and magnitude responses of IIR four-channel p-filter banks that generate tight frames. The p-filters are derived from the polynomial interpolating splines of orders 3, 4,5 and from the discrete splines of orders 6, 8, 10 and 12. Impulse responses of the p-filters $\mathbf{h}^s$ are the corresponding discrete-time framelets $\psi^s_{[1]}[k]$, $s = 0, 1, 2, 3$, of the first decomposition level. The framelets $\psi^s_{[1]}[k]$, $s = 0, 1, 2$, are symmetric, while the framelets $\psi^3_{[1]}[k]$ are antisymmetric. MRs of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ mirror each other and the same is true for the band-pass

pair $\mathbf{h}^2$ and $\mathbf{h}^3$. Note that, as the spline order is growing, the shapes of MRs of the p-filters $\mathbf{h}^0$ and $\mathbf{h}^1$ are approaching rectangles while MRs of the p-filters $\mathbf{h}^2$ and $\mathbf{h}^3$ are shrinking. In the caption to the figure, abbreviation IS means interpolating spline, DS means discrete spline.

Direct and inverse framelet transforms using four-channel p-filter banks are implemented by the MATLAB functions `fram34_dec_hm.m` and `fram34_rec_hm.m`, respectively. When some framelet transforms are repeatedly applied, it is advisable to use the functions `fram34_dec_hmwf.m` and `fram34_rec_hmwf.m`. The two-dimensional transforms are implemented by the MATLAB functions `fram34_dec2d_hm.m` and `fram34_rec2d_hm.m` (or `fram34_dec2d_hmwf.m` and `fram34_rec2d_hmwf.m`), respectively. Available frames are listed in the MATLAB code `list_frame4.m`. Figures 17.5, 17.6, 17.8, 17.7, 17.10, 17.11 and 17.12 were produced by the code `fram4_exam_P.m`. Figures 17.9 and 17.13 were produced by the code `fram4_exam_P7.m`.

References

1. A. Averbuch, V. Zheludev, T. Cohen, Interpolatory frames in signal space. IEEE Trans. Signal Process. **54**(6), 2126–2139 (2006)
2. L.M. Bregman, The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**(3), 200–217 (1967)
3. J. Cai, S. Osher, and Z. Shen. Split Bregman methods and frame based image restoration. Multiscale Model. Simul., 8(2):337–369, 2009/10.
4. J. Cai, Z. Shen, Framelet based deconvolution. J. Comput. Math. **28**(3), 289–308 (2010)
5. R.H. Chan, S.D. Riemenschneider, L. Shen, Z. Shen, High-resolution image reconstruction with displacement errors: a framelet approach. Internat. J. Imaging Systems Tech. **14**(3), 91–104 (2004)
6. R.H. Chan, S.D. Riemenschneider, L. Shen, Z. Shen, Tight frame: An efficient way for high-resolution image reconstruction. Appl. Comput. Harmon. Anal. **17**(1), 91–115 (2004)
7. C.K. Chui, W. He, Compactly supported tight frames associated with refinable functions. Appl. Comput. Harmon. Anal. **8**(3), 293–319 (2000)
8. Z. Cvetković, M. Vetterli, Oversampled filter banks. IEEE Trans. Signal Process. **46**(5), 1245–1255 (1998)
9. B. Dong, H. Ji, J. Li, Z. Shen, Y. Xu, Wavelet frame based blind image inpainting. Appl. Comput. Harmon. Anal. **32**(2), 268–279 (2012)
10. B. Dong, Z. Shen, Pseudo-splines, wavelets and framelets. Appl. Comput. Harmon. Anal. **22**(1), 78–104 (2007)
11. D. Gabor, Theory of communications. J. Inst. Electr. Eng. **93**, 429–457 (1946)
12. V.K. Goyal, J. Kovacevic, J.A. Kelner, Quantized frame expansions with erasures. Appl. Comput. Harmon. Anal. **10**(3), 203–233 (2001)
13. V.K. Goyal, M. Vetterli, N.T. Thao, Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis and algorithms. IEEE Trans. Inform. Theory **44**(1), 16–31 (1998)
14. J. Kovacevic, P.L. Dragotti, V.K. Goyal, Filter bank frame expansions with erasures. IEEE Trans. Inform. Theory **48**(6), 1439–1450 (2002)
15. A.P. Petukhov, Symmetric framelets. Constr. Approx. **19**(2), 309–328 (2003)
16. G. Polya, G. Szegö, *Aufgaben and Lehrsätze aus der Analysis*, vol. II (Springer, Berlin, 1971)
17. L. Shen, M. Papadakis, I.A. Kakadiaris, I. Konstantinidis, D. Kouri, D. Hoffman, Image denoising using a tight frame. IEEE Trans. Image Process. **15**(5), 1254–1263 (2006)

18. Z. Shen. Wavelet frames and image restorations. In R. Bhatia, editor, Proceedings of the International Congress of Mathematicians, Vol. IV, pages 2834–2863, New Delhi, 2010. Hindustan Book Agency.
19. W. Yin, S. Osher, D. Goldfarb, J. Darbon, Bregman iterative algorithms for l_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**(1), 143–168 (2008)
20. V. Zheludev, V. N. Malozemov, and A. B. Pevnyi. Filter banks and frames in the discrete periodic case. In N. N. Uraltseva, editor, Proceedings of the St. Petersburg Mathematical Society, Vol. XIV, volume 228 of Amer. Math. Soc. Transl., Ser. 2, pages 1–11, 2009.

Chapter 18
Application of Periodic Frames to Image Restoration

Abstract In this chapter, we present examples of image restoration using periodic frames. Images to be restored were degraded by blurring, aggravated by random noise and random loss of significant number of pixels. The images are transformed by periodic frames designed in Sects. 17.2 and 17.4, which are extended to the 2D setting in a standard tensor product way. In the presented experiments, performances of different tight and semi-tight frames are compared between each other in identical conditions.

18.1 Outline of the Restoration Scheme

Images are restored by the application of the *split Bregman iteration* (SBI) scheme [1] that uses the so-called *analysis-based* approach (see for example [2]).

Denote by $\mathbf{u} = \{u[\kappa, \nu]\}$ the original image array to be restored from the degraded array $\mathbf{f} = \mathbf{K}\,\mathbf{u} + \varepsilon$, where $\mathbf{K}$ denotes the operator of 2D discrete convolution of the array $\mathbf{u}$ with a kernel $\mathbf{k} = \{k[\kappa, \nu]\}$, and $\varepsilon = \left\{e_{k,n}\right\}$ is the random error array. $\bar{\mathbf{K}}$ denotes the conjugate operator of $\mathbf{K}$, which implements the discrete convolution with the transposed kernel $\mathbf{k}^T$. If some number of pixels are missing then the image $\mathbf{u}$ should be restored from the available data

$$\mathbf{P}_\Lambda\,\mathbf{f} = \mathbf{P}_\Lambda\,(\mathbf{K}\,\mathbf{u} + \varepsilon), \tag{18.1}$$

where the symbol $\mathbf{P}_\Lambda$ denotes the projection on the remaining set of pixels.

The solution scheme is based on the assumption that the original image $\mathbf{u}$ can be sparsely represented in a frame domain. Denote by $\tilde{\mathbf{F}}$ the operator of frame expansion of the image $\mathbf{u}$, where $\mathbf{C} \stackrel{\text{def}}{=} \tilde{\mathbf{F}}\,\mathbf{u}$, $\mathbf{C} = \{c[\kappa, \nu]\}$, is the set of the frame transform coefficients. Denote by $\mathbf{F}$ the operator of reconstruction of the image $\mathbf{u}$ from the set of the frame coefficients. We get $\mathbf{F}\,\mathbf{C} = \mathbf{u}$, $\mathbf{F}\,\tilde{\mathbf{F}} = \mathbf{I}$, where $\mathbf{I}$ is the identity operator.

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4_18,

An approximated solution to Eq. (18.1) is derived via minimization of the functional

$$\min_{u} \frac{1}{2} \|\mathbf{P}_{\Lambda} (\mathbf{K}\,\mathbf{u} - f)\|_2^2 + \lambda \left\| \tilde{\mathbf{F}}\,\mathbf{u} \right\|_1, \tag{18.2}$$

where $\|\cdot\|_1$ and $\|\cdot\|_2$ are the l_1 and the l_2 norms of the sequences, respectively. If $\mathbf{x} = \{x[\kappa, \nu]\}$, $\kappa = 0, ..., k$, $\nu = 0, ..., n$, then

$$\|\mathbf{x}\|_1 \stackrel{\text{def}}{=} \sum_{\kappa=0}^{k-1} \sum_{\nu=0}^{n-1} |x[\kappa, \nu]|, \quad \|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{\kappa=0}^{k-1} \sum_{\nu=0}^{n-1} |x[\kappa, \nu]|^2}.$$

Denote by $\mathbf{T}_{\vartheta}$ the operator of soft thresholding:

$$\mathbf{T}_{\vartheta}\,\mathbf{x} = \{x_{\vartheta}[\kappa, \nu]\}, \quad x_{\vartheta}[\kappa, \nu] \stackrel{\text{def}}{=} \operatorname{sgn}(x[\kappa, \nu]) \max\{0, |x[\kappa, \nu]| - \vartheta\}.$$

Following [2], we solve the minimization problem in Eq. (18.2) by an iterative algorithm. We begin with the initialization $\mathbf{u}^0 = 0$, $\mathbf{d}^0 = \mathbf{b}^0 = 0$. Then,

$$\begin{aligned} \mathbf{u}^{k+1} &:= (\bar{\mathbf{K}}\,\mathbf{P}_{\Lambda}\,\mathbf{K} + \mu\,\mathbf{I})\,\mathbf{u} = \bar{\mathbf{K}}\,\mathbf{P}_{\Lambda}\,\mathbf{f} + \mu\,\mathbf{F}\,(\mathbf{d}^k - \mathbf{b}^k), \\ \mathbf{d}^{k+1} &= \mathbf{T}_{\lambda/\mu}(\tilde{\mathbf{F}}\,\mathbf{u}^{k+1} + \mathbf{b}^k), \\ \mathbf{b}^{k+1} &= \mathbf{b}^k + (\tilde{\mathbf{F}}\,\mathbf{u}^{k+1} - \mathbf{d}^{k+1}). \end{aligned} \tag{18.3}$$

The linear system in the first line of Eq. (18.3) is solved by the application of the *conjugate gradient* algorithm. The operations in the second and third lines are straightforward. The choice of the parameters λ and μ depends on experimental conditions.

18.2 Numerical Examples

The restoration algorithms were applied to the "Window", "Barbara", "Boats", "Lena" and "Fingerprint" images. These images were blurred by convolution with either the motion kernel or the Gaussian kernels and were degraded by loss of a large number of pixels. In some experiments, the degradation was aggravated by addition of zero mean random noise.

In the presented examples, we compare the performance of tight frames (TF) to that of semi-tight frames (STF) derived from different polynomial and discrete splines. The available frames using 3-channel p-filter banks are listed in Table 18.1, while frames using 4-channel p-filter banks are listed in Table 18.2.

The proximity between an image $\tilde{\mathbf{u}}$ and the original image $\mathbf{u}$ is evaluated visually and by the Peak-Signal-to-Noise ratio (PSNR) (Eq. 6.42).

Table 18.1 List of spline-based frames using 3-channel p-filter banks

$T^3_{1,0}$	Linear IS TF,	$T^3_{2,0}$	QU IS TF,
$T^3_{3,0}$	CU IS TF,	$S^3_{3,1}$	CU IS STF (SY, IIR),
$S^3_{3,2}$	CU IS STF (SY, FIR),	$S^3_{3,3}$	CU IS STF (ASY),
$T^3_{7,0}$	QU QIS TF,	$S^3_{7,1}$	QU QIS STF (SY),
$S^3_{7,2}$	QU QIS STF (ASY),	$S^3_{7,3}$	QU QIS STF (4 LDVMs in analysis framelet)
$T^3_{6,0}$	DS of 6-th order TF,	$S^3_{6,1}$	DS of 6-th order STF (SY),
$T^3_{8,0}$	DS of 8-th order TF,	$T^3_{12,0}$	DS of 12-th order TF.

IS interpolating spline, *QIS* quasi-interpolating spline, *DS* discrete spline, *TF* tight frame, *STF* semi-tight frame, *LPF* low-pass p-filter, *SY* symmetric, *ASY* antisymmetric, *QU* quadratic, *CU* cubic

Table 18.2 List of spline-based frames using 4-channel p-filter banks

$T^4_{1,0}$	Linear IS TF,	$T^4_{2,0}$	QU IS TF,
$S^4_{2,1}$	QU IS STF (ASY)	$T^4_{3,0}$	CU IS TF,
$S^4_{3,1}$	CU IS STF (SY, FIR),	$S^4_{3,2}$	CU IS STF (ASY),
$T^4_{4,0}$	4-th degree IS TF,	$S^4_{4,1}$	4-th degree IS STF (SY),
$S^4_{4,2}$	4-th degree IS STF (ASY),	$T^4_{5,0}$	QU QIS TF (interp. LPF),
$S^4_{5,1}$	QU QIS STF (SY, interp. LPF),	$S^4_{5,2}$	QU QIS STF (ASY, interp. LPF),
$T^4_{6,0}$	QU QIS TF (non-interp. LPF),	$S^4_{6,1}$	QU QIS STF (SY, non-interp. LPF),
$S^4_{6,2}$	QU QIS STF (ASY, non-interp. LPF),	$T^4_{7,0}$	Upgraded QU QIS TF,
$S^4_{7,1}$	Upgraded QU QIS STF (SY, non-interp. LPF),	$T^4_{8,0}$	Pseudo-spline TF,
$S^4_{8,1}$	Pseudo-spline STF (SY),	$S^4_{8,2}$	Pseudo-spline STF (ASY),
$T^4_{60,0}$	DS of 6-th order TF,	$T^4_{80,0}$	DS of 8-th order TF.
$T^4_{100,0}$	DS of 10-th order TF,	$T^4_{120,0}$	DS of 12-th order TF.

IS interpolating spline, *QIS* quasi-interpolating spline, *DS* discrete spline, *TF* tight frame, *STF* semi-tight frame, *LPF* low-pass p-filter, *SY* symmetric, *ASY* antisymmetric, *QU* quadratic, *CU* cubic

18.2.1 Restoration Experiments with the "Window" Image

This image was taken from [2], where results of two restoration experiments were cited. The experiments described below were carried out in the conditions similar to the conditions cited in [2].

Experiment 1.1: Image blurred and pixels missing

The image was blurred by convolution with the motion kernel (MATLAB function `fspecial`('motion',15,45)) and its PSNR becomes 24.15 dB. Then, 30 % of the pixels were randomly removed. This reduces the PSNR to 11.61 dB. No random noise was added. The image was restored by 50 SBI, using the parameters $\lambda = 0.001,\ \mu = 0.001$ in Eq. (18.3). The conjugate gradient solver used 100 iterations. Tight and semi-tight frames listed in Tables 18.1 and 18.2 were tested. Decomposition is implemented down to the fifth level. The restored PSNR results are illustrated by the diagram in Fig. 18.1. The bars in the diagram are arranged in the order given in Table 18.3.

The best PSNR $= 44.09$ dB result for the restored image was produced by the tight frame $\mathbf{T}^4_{8,0}$ derived from the pseudo-spline. A close PSNR $= 44.0$ dB value

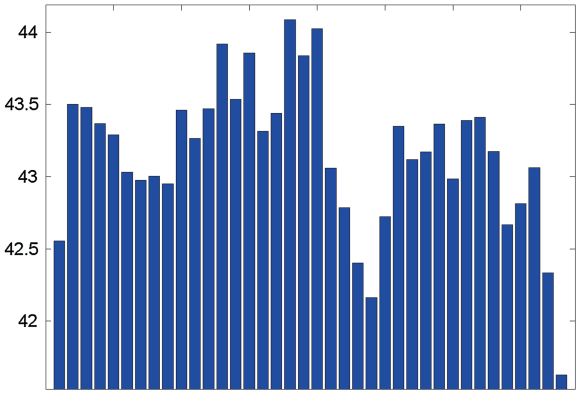


Fig. 18.1 PSNR results for Experiment 1.1. The best PSNR $= 44.09$ dB is achieved by the frame $\mathbf{T}_{8,0}^4$

was produced by the semi-tight frame $\mathbf{S}_{8,2}^4$ derived from the pseudo-spline with antisymmetric factorization of the sequence $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ (Sect. 17.4.1). Figure 18.2 displays the restoration result. Visually, the restored image hardly can be distinguished from the original one.

Experiment 1.2: Image blurred and noised and pixels missing

The image was blurred by convolution with the motion kernel (MATLAB function `fspecial`('motion',15,45)) and a weak random zero-mean Gaussian noise, whose STD is $\sigma = 5$, was added. The PSNR was 23.74 dB. Then, 30 % of the pixels were randomly removed. This reduces the PSNR to 11.59 dB. The image was restored by 30 SBI, using the parameters $\lambda = 0.3$, $\mu = 0.1$ in Eq. (18.3). The conjugate gradient solver used 10 iterations. Tight and semi-tight frames listed in Tables 18.1 and 18.2 were tested. Decomposition is implemented down to the second level. The restored PSNR results are illustrated in Fig. 18.3 (see Table 18.3).

As in the previous experiment, the best PSNR $= 28.57$ dB result for the restored image was produced by the tight frame $\mathbf{T}_{8,0}^4$ derived from the pseudo-spline. The value PSNR $= 28.56$ dB was achieved by the semi-tight frame $\mathbf{S}_{8,2}^4$ derived from the pseudo-spline with antisymmetric factorization of the sequence $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ (Sect. 17.4.1). Frames derived from the quadratic quasi-interpolating spline (non-interpolating loq-pass p-filter) produced very close PSNR results. Figure 18.4 displays the restoration result. Noise is completely removed and the image is effectively deblurred.

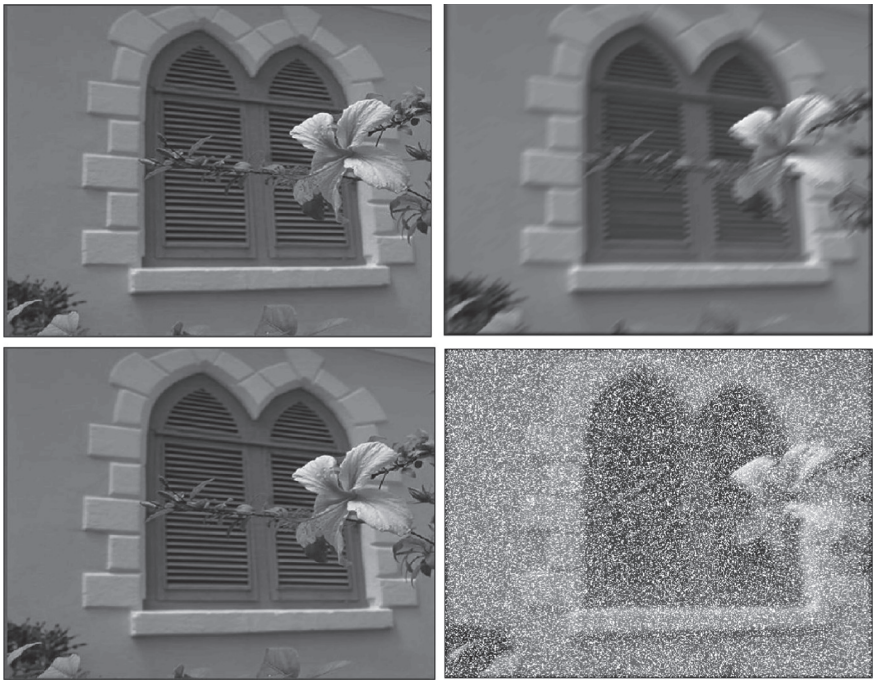


Fig. 18.2 *Top left* Source input—the "Window" image. *Top right* Blurred. PSNR $= 24.15\,\text{dB}$. *Bottom right* After random removal of 30 % of its pixels. PSNR $= 11.61\,\text{dB}$. *Bottom left* The image restored by the frame $\mathbf{T}^4_{8,0}$. PSNR $= 44.09\,\text{dB}$

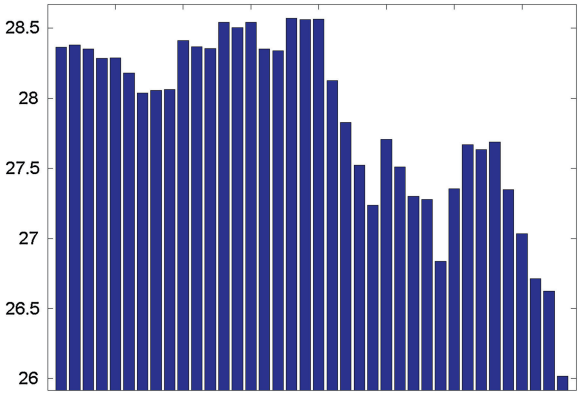


Fig. 18.3 PSNR results for Experiment 1.2. The best PSNR $= 28.57\,\text{dB}$ is achieved by the frame $\mathbf{T}^4_{8,0}$

Table 18.3 Order of bars in the diagrams

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$T^4_{1,0}$	$T^4_{2,0}$	$S^4_{2,1}$	$T^4_{3,0}$	$S^4_{3,1}$	$S^4_{3,2}$	$T^4_{4,0}$	$S^4_{4,1}$	$S^4_{4,2}$	$T^4_{5,0}$	$S^4_{5,1}$	$S^4_{5,2}$	$T^4_{6,0}$	$S^4_{6,1}$	$S^4_{6,2}$	$T^4_{7,0}$	$S^4_{7,1}$	$T^4_{8,0}$	$S^4_{8,1}$
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
$S^4_{8,2}$	$T^4_{60,0}$	$T^4_{80,0}$	$T^4_{100,0}$	$T^4_{120,0}$	$T^3_{1,0}$	$T^3_{2,0}$	$T^3_{3,0}$	$S^3_{3,1}$	$S^3_{3,2}$	$S^3_{3,3}$	$T^3_{7,0}$	$S^3_{7,1}$	$S^3_{7,2}$	$S^3_{7,3}$	$T^3_{6,0}$	$S^3_{6,1}$	$T^3_{8,0}$	$T^3_{12,0}$

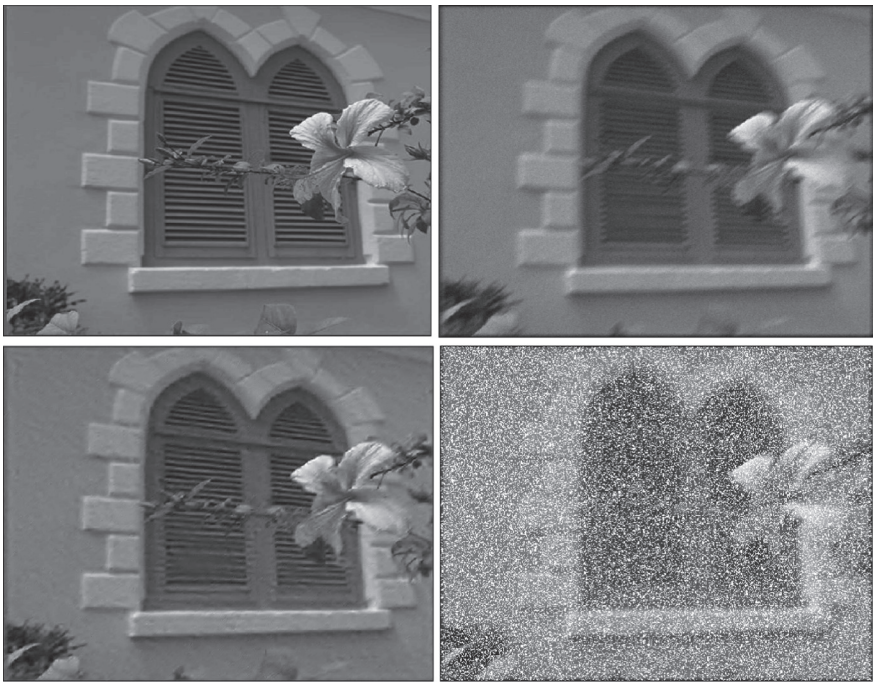


Fig. 18.4 *Top left* Source input—the "Window" image. *Top right* Blurred and noised. PSNR = 23.74 dB. *Bottom right* After random removal of 30 % of its pixels. PSNR = 11.59 dB. *Bottom left* The image restored by the frame $\mathbf{T}^4_{8,0}$. PSNR = 28.57 dB

18.2.2 Restoration Experiments with the "Barbara" Image

The "Barbara" image has fine texture, which vanishes when blurring and should be restored despite a significant loss of pixels.

Experiment 2.1: Image blurred and pixels missing

In these experiments, the "Barbara" image was restored after it was blurred by a convolution with the Gaussian kernel (MATLAB function `fspecial`('gaussian', [5 5],5)) and its PSNR became 23.32 dB. Then, 50 % of its pixels were randomly removed. This reduces the PSNR to 9.34 dB. Random noise was not added. The image was restored by 50 SBI using the parameters $\lambda = 0.001$, $\mu = 0.001$ in

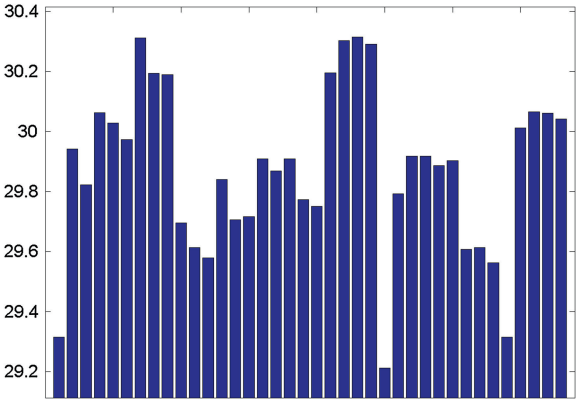


Fig. 18.5 PSNR results for Experiment 2.1. The best PSNR = 30.33 dB is achieved by the frame $\mathbf{T}^4_{100,0}$

Eq. (18.3). The conjugate gradient solver used 100 iterations. Tight and semi-tight frames listed in Tables 18.1 and 18.2 were tested. Decomposition is implemented down to the second level. The restored PSNR results are illustrated in Fig. 18.5 (see Table 18.3).

The best PSNR = 30.32 dB result for the restored image was produced by the tight frame $\mathbf{T}^4_{100,0}$ derived from the discrete spline of tenth order. Almost the same value PSNR = 30.31 dB was achieved by the tight frame $\mathbf{T}^4_{4,0}$ derived from the polynomial interpolating spline of fourth degree. The frames derived from the discrete splines of eighth and twelfth order produced very close PSNR results. Figure 18.6 displays the restoration result. The image is deblurred and the fine texture is restored almost completely.

Experiment 2.2: Image blurred, noised and pixels missing

In these experiments, the "Barbara" image was restored after being blurred by a convolution with the Gaussian kernel (MATLAB function `fspecial`('gaussian', [5 5],5)) and corrupted by a moderate random zero-mean Gaussian noise, whose STD is $\sigma = 10$. The PSNR was 22.08 dB. Then, 50 % of its pixels were randomly removed. This reduces the PSNR to 9.31 dB. The image was restored by 50 SBI, using the parameters $\lambda = 0.8$, $\mu = 0.03$ in Eq. (18.3). The conjugate gradient solver used 15 iterations. Tight and semi-tight frames listed in Tables 18.1 and 18.2 were tested. Decomposition is implemented down to the fifth level. The restored PSNR results are illustrated in Fig. 18.7 (see Table 18.3).

The best PSNR = 24.1886 dB result for the restored image was produced by the semi-tight frame $\mathbf{S}^4_{8,2}$ derived from the pseudo-spline. Almost the same value PSNR = 24.1853 dB was achieved by the semi-tight frame $\mathbf{S}^4_{6,2}$ derived from the quadratic quasi-interpolating spline (non-interpolating low-pass p-filter). Figure 18.8 displays the restoration result. The image is deblurred to some extent, noise is removed and the fine texture is restored only partially.

Fig. 18.6 *Top left* Source input—"Barbara" image. *Top right* Blurred, PSNR $= 23.32$ dB. *Bottom right* After random removal of 50 % of its pixels. PSNR $= 9.34$ dB. *Bottom left* The image restored by the frame $\mathbf{T}^4_{100,0}$. PSNR $= 30.32$ dB

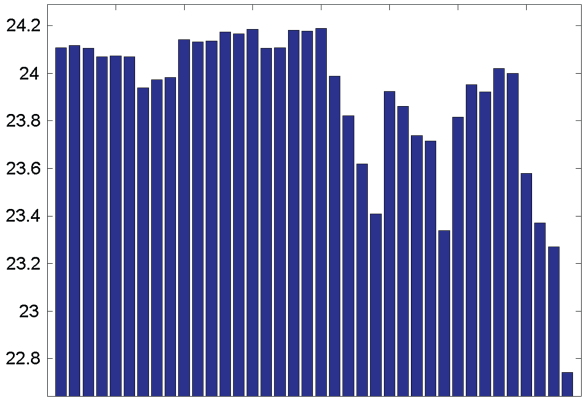


Fig. 18.7 PSNR results for Experiment 2.2. The best PSNR $= 24.1886$ dB is achieved by the frame $\mathbf{S}^4_{8,2}$

Fig. 18.8 *Top left* Source input—the "Barbara" image. *Top right* Blurred and noised, PSNR = 22.08 dB. *Bottom right* After random removal of 50 % of its pixels. PSNR = 9.31 dB. *Bottom left* The image restored by the frame $\mathbf{S}^4_{8,2}$. PSNR = 24.19 dB

18.2.3 Restoration Experiments with the "Boats" Image

Experiment 3.1: Image blurred and pixels missing

In these experiments, the "Boats" image was restored after it was blurred by a convolution with the motion kernel (MATLAB function `fspecial('motion',15,45)`), and its PSNR became 22.62 dB. Then, 70 % of its pixels were randomly removed. This reduces the PSNR to 6.37 dB. Random noise was not added. The image was restored by 50 SBI, using the parameters $\lambda = 0.005,\ \mu = 0.001$ in Eq. (18.3). The conjugate gradient solver used 40 iterations. Tight and semi-tight frames listed in Tables 18.1 and 18.2 were tested. Decomposition is implemented down to the first level. The restored PSNR results are illustrated in Fig. 18.9 (see Table 18.3).

As in Experiment 2.2 the best PSNR = 31.39 dB result for the restored image was produced by the semi-tight frame $\mathbf{S}^4_{8,2}$ derived from the pseudo-spline. Almost the same value PSNR = 31.32 dB was achieved by the semi-tight frame $\mathbf{S}^4_{6,2}$ derived from the quadratic quasi-interpolating spline (non-interpolating low-pass p-filter). Frames derived from the discrete splines of eighth and twelfth order produced very close PSNR results. Figure 18.10 displays the restoration result. The image is deblurred and the fine texture is restored almost completely.

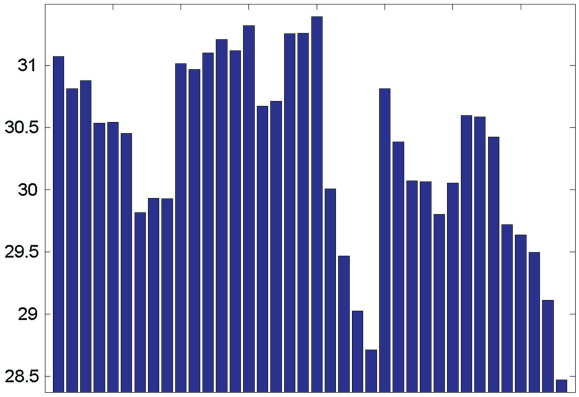


Fig. 18.9 PSNR results for Experiment 3.1. The best PSNR $=$ 31.39 dB is achieved by the frame $\mathbf{S}^4_{8,2}$

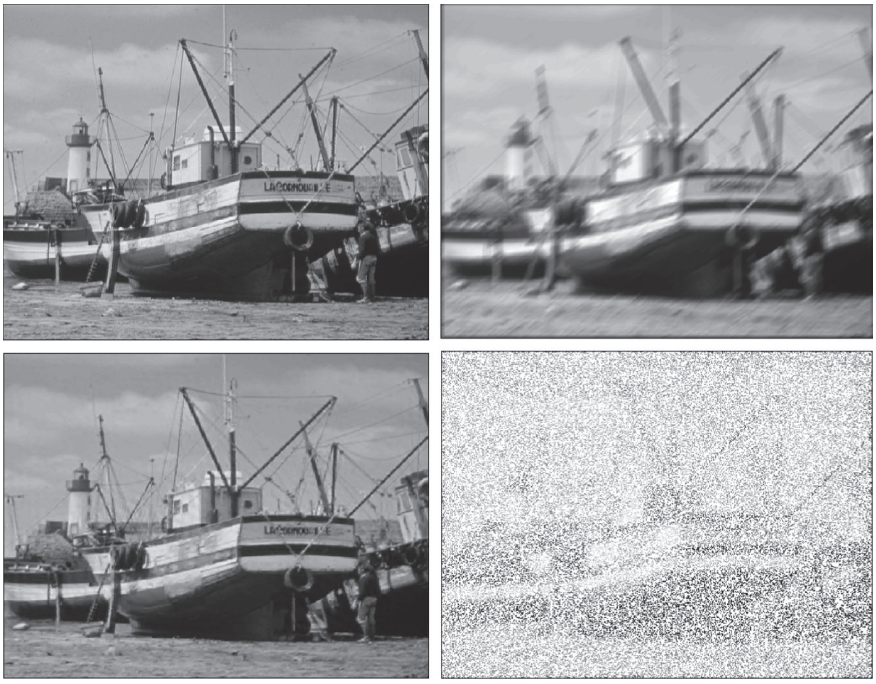


Fig. 18.10 *Top left* Source input—the "Boats" image. *Top right* Blurred, PSNR $=$ 22.62 dB. *Bottom right* After random removal of 70 % of its pixels. PSNR $=$ 6.37 dB. *Bottom left* The image restored by the frame $\mathbf{S}^4_{8,2}$. PSNR $=$ 31.39 dB

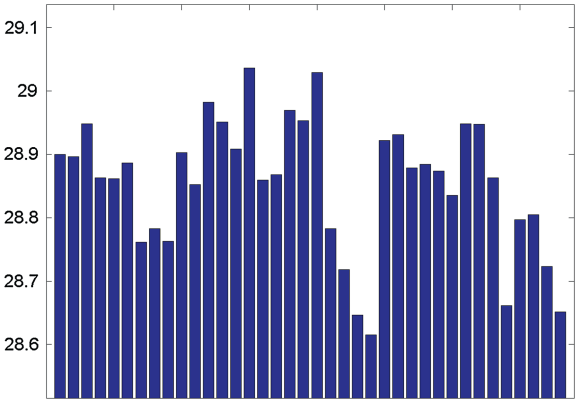


Fig. 18.11 PSNR results for Experiment 4.1. The best PSNR = 29.04 dB is achieved by the frame $\mathbf{S}^4_{6,2}$

18.2.4 Restoration Experiments with the "Lena" Image

Experiment 4.1: Image blurred and distorted by curves

The source image was severely blurred by convolution with a Gaussian kernel (MATLAB function `fspecial`('gaussian',[12,12],12)) (PSNR = 23.64 dB). The blurred image was distorted by randomly drawn curves (PSNR = 16.44 dB). Noise was not added. The image is restored by 50 SBI with the parameters $\lambda = 0.06$ and $\mu = 0.17$ by the tight and semi-tight frames listed in Tables 18.1 and 18.2. Decomposition was implemented down to the fifth level. The conjugate gradient solver used 11 iterations. The restored PSNR results are illustrated in Fig. 18.11 (see Table 18.3).

The best PSNR = 29.04 dB result for the restored image was produced by the semi-tight frame $\mathbf{S}^4_{6,2}$ derived from the quadratic quasi-interpolating spline (non-interpolating low-pass p-filter). Almost the same value PSNR = 29.04 dB was achieved by the semi-tight frame $\mathbf{S}^4_{8,2}$ derived from the pseudo-spline. Figure 18.12 displays the restoration result. The image is deblurred and the distorting curves are completely removed.

18.2.5 Restoration Experiments with the "Fingerprint" Image

Experiment 5.1: Image blurred, noised and pixels missing

Unlike previous experiments, the "Fingerprint" image was affected by a strong zero-mean white noise with STD $\sigma = 20$ after being blurred by convolution with the Gaussian kernel (MATLAB function `fspecial` ('gaussian',[5 5],5)) (PSNR = 19.66 dB). Then, 50 % of its pixels were randomly removed and this produced PSNR = 7.45 dB. The image was restored by 40 SBI with the parameters $\lambda = 2.3$ and $\mu = 0.35$ using the tight and semi-tight frames listed in Tables 18.1 and 18.2. Decomposition is implemented down to the fifth level. The conjugate gradient

Fig. 18.12 *Top left* Source input—the "Lena" image. *Top right* Blurred, PSNR $= 23.64$ dB. *Bottom right* Distorted by random curves. PSNR $= 16.44$ dB. Bottom left: The image restored by the frame $\mathbf{S}^4_{6,2}$. PSNR $= 29.04$ dB

solver used 10 iterations. The restored PSNR results are illustrated in Fig. 18.13 (see Table 18.3).

The best PSNR $= 23.78$ dB result for the restored image was produced by the tight frame $\mathbf{T}^4_{80,0}$ derived from the discrete spline of eighth order. Almost the same value PSNR $= 23.76$ dB was achieved by the tight frame $\mathbf{T}^4_{4,0}$ derived from the polynomial interpolating spline of fourth degree. The frames derived from the discrete splines of tenth and twelfth order produced very close PSNR results as did the semi-tight frames derived from the polynomial interpolating spline of fourth degree. Figure 18.14 displays the restoration result. The image is deblurred, noise is almost completely removed and the texture is restored.

18.3 Comments on the Experiments

In the above experiments, spline-based tight and semi-tight frames proved to be highly efficient for image restoration. It is important that the computational cost does not depend on the spline order. This makes the framelets with high number of

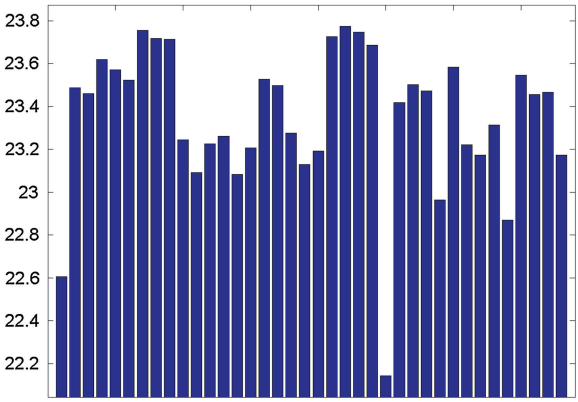


Fig. 18.13 PSNR results for Experiment 5.1. The best PSNR $= 23.78$ dB is achieved by the frame $\mathbf{T}^4_{80,2}$

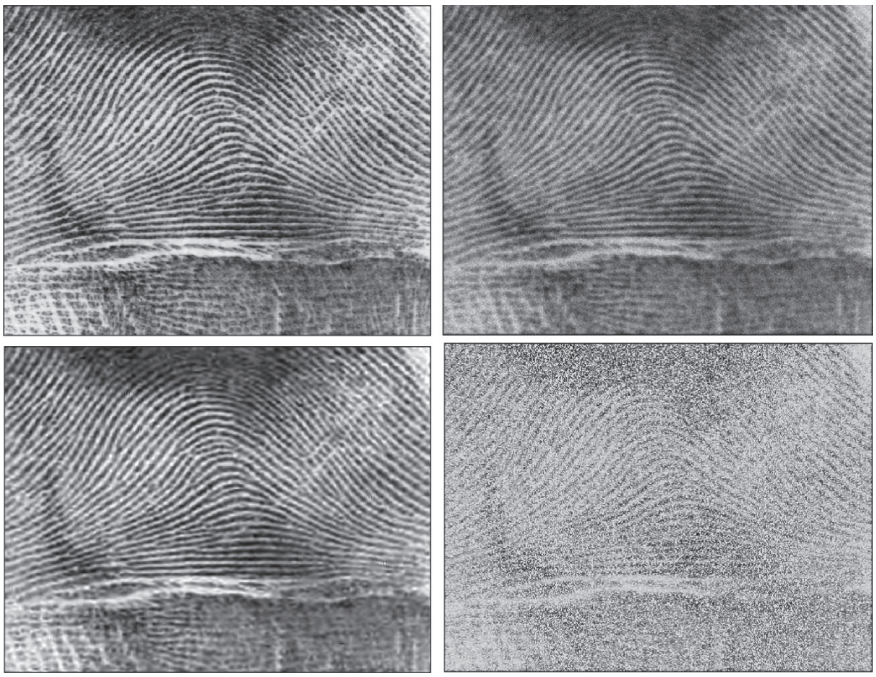


Fig. 18.14 *Top left* Source input—"Fingerprint" image. *Top right* Blurred and noised, PSNR $=$ 19.66 dB. *Bottom right* After random removal of 50 % of its pixels. PSNR $= 7.45$ dB. *Bottom left* The image restored by the frame $\mathbf{T}^4_{80,0}$. PSNR $= 23.78$ dB

LDVMs and infinite impulse response suitable for utilization in implementation. In some cases, such as the Experiment 2.1 with "Barbara" and Experiment 5.1 with "Fingerprint", the best results were produced by tight frames derived from higher order discrete splines and from the interpolating spline of fourth degree. Note that both images have fine texture, which is satisfactorily restored after strong blurring, corrupting by noise ("Fingerprint") and subsequent random removal of 50 % of their pixels.

When image distortion is not strong, as in Experiment 1.1 with the "Window" image, it is restored close to perfection. The images "Lena", which was strongly blurred, and "Boats", where 70 % of its pixels were missing, were successfully restored. For these three images, which are relatively smooth, the best results were produced by the semi-tight frames, which are derived from the quasi-interpolating quadratic spline and from the pseudo-spline. The low-pass p-filters in the generating p-filter banks are non-interpolating and have relatively long IRs (nine taps). Note that in most cases, the semi-tight frames performed better than their tight counterparts.

Typically, increase in the SBI number above 50 does not contribute into the restoration quality. In some cases, as in the "Window" and "Barbara" experiments, a large number of iterations of the conjugate gradient solver increased the restoration quality. However, in the experiments with the strongly degraded "Lena" and "Fingerprint" images, increase in the number of iterations above 12 depleted the quality.

Choice of the regularization parameters λ and μ is of crucial importance. The restoration quality is sensitive to them. Even a small change in the parameters significantly affects it. The number of decomposition levels is also important.

Generally, frames using four-channel p-filter banks were advantageous over the three-channel frames, but the results do not differ significantly.

The experiments on image restoration by the Bregman iterations using the spline-based frames are implemented by the MATLAB code `run_Bregman.m`. The performance comparison is carried out by the code `compare_frames.m`. The comparison results are being saved in the file `TAB.txt`.

References

1. T. Goldstein, S. Osher, The split Bregman method for L1-regularized problems. SIAM J. Imaging Sci. **2**(2), 323–343 (2009)
2. H. Ji, Z. Shen, Y. Xu, Wavelet based restoration of images with missing or damaged pixels. East Asian J. Appl. Math. **1**(2), 108–131 (2011)

Appendix
Guide to SplineSoftP

The toolbox contains MATLAB codes, which implement the spline-based methods described in the above chapters. It comprises 13 folders:

1. SplineP,
2. SubDivP,
3. ConvolutionP,
4. RMPursuitP,
5. Block_basP,
6. W_packetsP,
7. Discr_SplineP,
8. Discr_SplineWaveletP,
9. Discr_SplineRMPursuitP,
10. B_waveletsP,
11. FramesP,
12. Bregman_iteration,
13. Utilities,
14. Store.

SplineP

This folder is associated with Chaps. 3 and 4.

`bsplP.m` This function computes values of the B-splines $B^p(t)$ of any order using Eq. 3.14.

`GensplConsP.m` The code displays generators of even-order spline spaces and so also their SHA spectra.

`spligenP.m` The function designs a single generator of an even-order spline space, using the binary subdivision. For subdivision it calls the function `interssplliPt.m`.

`GensplConsP3.m` The code displays generators of odd-order spline spaces.

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4,

`spligenP3.m` The function designs a single generator of an odd-order spline space, using the ternary subdivision. For subdivision it calls the function `intersplliPt.m`.

`bsplsP.m` The code displays the B-splines.

`bsplFP.m` The code displays Fourier spectra of the B-splines.

`bspuvP.m` The function calculates samples of B-splines of multiple orders at integer and semi-integer points and saves them in the file `BSUV.mat`.

`juviP.m` The function designs characteristic sequence of the space of N-periodic splines of a given order, which is the DFT of the B-spline sampled at integer points, and computes the DFT of the B-spline sampled at semi-integer points.

SubDivP

This folder is associated with Chap. 6.

`intersplliPt.m` This function implements binary/ternary N-periodic spline subdivision (any spline order for ternary subdivision, even spline order for binary subdivision).

`intersplli2DPt.m` The function implements binary/ternary 2D spline subdivision using the function `intersplliPt.m`. The subdivision can be either of the same type in both directions or ternary in one direction and binary in the other.

`smoothsplaPt.m` This function implements binary/ternary subdivision of a smoothing spline (any spline order for ternary subdivisionusing the function `intersplliPt.m` (any spline order for ternary subdivision,even spline order for binary subdivision).

`smooth_spline2DPt.m` The function implements binary/ternary subdivision of a 2D smoothing spline using the function `intersplli2DPt.m`. The subdivision can be either of the same type in both directions or ternary in one direction and binary in the other.

`defroPSm.m` The function finds the optimal regularization parameter for smoothing splines.

ConvolutionP

This folder is associated with Chap. 7.

`conv_splaP.m` This function implements the spline-based Tikhonov deconvolution algorithm, using samples of an input signal at integer points. The input signal is blurred with a kernel and/or corrupted by noise. Restoration of the samples is followed by the binary spline subdivision.

`convsplaP2.m` Does the same for two-dimensional signals.

`buikerP.m` This function is modeling blurring kernels for 2D convolution and the heat equation.

`defroP_conv.m` The function finds the optimal regularization parameter for the 1D and 2D Tikhonov deconvolution and inversion of the heat equation.

`heatsplaP2.m` This function implements the 2D spline-based Tikhonov algorithm for inversion of the heat equation, using samples of an input signal at integer points. The input signal is blurred with a kernel and/or corrupted by noise. Restoration of the samples of the initial temperature distribution is followed by the 2D binary spline subdivision.

`ETTA_P.m` This function is preparing coefficients for inversion of the heat equation for the collocation and the difference approximation.

W_packetsP

This folder is associated with Chaps. 8 and 9.

`spl_Wv_analP.m` This function implements the 1D wavelet transforms of a row signal or an array of row signals. The transforms are using the spline-based p-filters as it is described in Chap. 9.

`spl_WP_analP.m` This function implements the 1D wavelet packet transforms of a row signal or an array of row signals, using the spline-based p-filters as it is described in Chap. 9.

`spl_WP_analP2D.m` Does the same for two-dimensional signals.

`samp_spl_WP_analP.m` This function implements the 1D wavelet packet transforms of an array, where each row consists of grid samples of a spline. It produces the spline's coordinates in the wavelet packets bases as it is described in Chap. 8.

`samp_spl_WP_analP2D.m` Does the same for two-dimensional splines.

`wP_downP.m` This function implements one step of the spline wavelet packet transform of a signal using the transform matrix.

`wP_down_2DP.m` Does the same for two-dimensional signals.

`matr_transP.m` The function prepares the transform matrices for the direct and inverse wavelet packets transforms.

`spl_Wv_syntP.m` This function implements synthesis of a row signal or an array of row signals from the set of the spline wavelet transform coefficients.

`spl_Wv_syntP_2D.m` Does the same for two-dimensional signals.

`horizontal_WP_synthP.m` This function implements synthesis of a row signal or an array of row signals from the set of the spline wavelet packet transform coefficients from a single decomposition level.

`List_synthP.m` This function implements synthesis of a row signal or an array of row signals from the set of the spline wavelet packet transform coefficients. The set is defined by a list of wavelet packets, which generate a basis of the signal space.

`List_synthP_2D.m` Does the same for two-dimensional signals.
`WP_BB_Listree1D_P.m` This function prepares the list of wavelet packets, which generate a Best Basis tree.
`WP_BB_Listree2_D_P.m` Does the same in the two-dimensional case.
`wp_genP.m` This function designs spline wavelet packets of various orders using the binary subdivision.
`wavs_coefP.m` This function computes SHA spectra of wavelet packets of various orders and their grid samples.
`lohifilP.m` This function designs frequency responses of the low-pass s and the high-pass d spline wavelet filters.

RMPursuitP

This folder is associated with Chap. 10.

`RMP_runP.m` This code launches the interactive process of the Regularized Matching Pursuit of columns of an image, which are blurred either by a Gaussian kernel or by time evolution in the heat equation, and/or are corrupted by additive noise.
`loadParametersMP_P.m` This function supports the user interface for setting the processing parameters.
`buildInputFunctions_RMP.m` This function prepares the orinal image, the blurred image, noised blurred input, the DFT of the blurring kernel and an additive noise.
`wavs_theta_coefP.m` This function calculates the SHA spectra of wavelet packets for the approximating dictionary and the DFT of testing waveforms (wavelet packets convolved with the kernel).
`tnorrP.m` This function calculates norms of testing waveforms.
`nois_model.m` This function emulates the additive noise by analysis of the input signals, as is described in Sect. 10.2.2.
`purs_optP.m` This function computes the oblique projection coefficients of the remainder. In addition, it is finding an optimal waveform, and determines its position.
`rm_pursP.m` This function implements the Regularized Matching Pursuit of a row signal. It does the main job in the folder.
`RMP_displayP.m` This function displays the result of the Regularized Matching Pursuit.
`build_kernP.m` This function prepares the sampled kernel and its DFT either for the deconvolution or for inversion of the heat equation with either difference or collocation approximation.

Block_basP

This folder is associated with Chap. 11.

`Block_runP.m` This code launches the interactive process of the 2D block-based inversion of the heat equation and the deconvolution and compares results with the results from the global Tikhonov algorithm.

`loadParametersB_P.m` This function supports the user interface for setting the processing parameters.

`buildInputFunctions_BP.m` This function prepares of the orinal image, the blurred image, noised blurred input, the DFT of the blurring kernel and an additive noise.

`weteviP.m` This function prepares auxiliary coefficients to be used in the global and block-wise Tikhonov solutions.

`tikh_dec.m` This function produces a regularized Tikhonov solution to either the deconvolution or the inversion problem.

`defro_PN.m` This function determines an optimal regularization parameter.

`deconExecute_BbpP.m` This function implements the Block Pursuit algorithm, using the functions `wavs_coefP.m` and `spl_WP_anaP2D.m` from the *W_packets* folder.

`deconExecute_BbbP.m` This function implements the block-based Best Basis algorithm, using the functions `wavs_coefP.m` and `spl_WP_anaP2D.m` from the *W_packets* folder.

`onePacketDecon_PN.m` This function finds the partial Tikhonov solution in a subspace defined by a single block of the wavelet packet coefficients.

`WP_BB_Listree2D_P.m` This function prepares the list of wavelet packets, which generate a best basis tree for the input image. The wavelet packets in the list are sorted according to the energies of the respective blocks of the transform coefficients.

`sort_list_P.m` This function is sorting the wavelet packets included in the Best Basis list according to the norms of their convolutions with the blurring kernel.

`thresh_det.m` This function is setting threshold for the Block Pursuit algorithm.

Discr_SplineP

This folder is associated with Chap. 13.

`per_dbspP.m` This function computes values of the discrete periodic B-spline, its DFT, its samples on the sparse grid and the characteristic sequence of the respective discrete spline space.

`D_gen_cons.m` This function designs generators of the discrete spline spaces, their DFT and samples on the sparse grid.

`D_smoothsplaP.m` This function computes values of the discrete periodic spline, which is interpolating or smoothing a signal on a sparse grid.

`D_smoothsplaP_2d.m` Does the same in the two-dimensional case.

`D_defroP.m` This function determines an optimal regularization parameter for the discrete smoothing splines.

Discr_SplineWaveletP

This folder is associated with Chap. 14.

`D_wav_analP.m` This function implements the 1D discrete spline wavelet transforms of a row signal or an array of row signals.

`D_wav_anal_2dP.m` Does the same for two-dimensional signals.

`D_WP_analP.m` This function implements the 1D discrete spline wavelet packet transforms of a row signal or an array of row signals, using the function `wP_downP.m` from the *W_packetsP* folder.

`D_WP_anal_2DP.m` Does the same for two-dimensional signals, using the function `wP_down_2DP.m` from the *W_packetsP* folder.

`matr_transP.m` The function prepares the transform matrices for the direct and inverse wavelet packets transforms.

`D_wav_syntP.m` This function implements synthesis of a row signal or an array of row signals from the set of the discrete spline wavelet transform coefficients.

`D_wav_synt_2dP.m` Does the same for two-dimensional signals.

`D_hor_WP_synthP.m` This function implements synthesis of a row signal or an array of row signals from the set of the discrete spline wavelet packet transform coefficients from a single decomposition level.

`D_hor_WP_synth_2DP.m` Does the same for two-dimensional signals.

`D_List_synthP.m` This function implements synthesis of a row signal or an array of row signals from the set of the discrete spline wavelet packet transform coefficients. The set is defined by a list of wavelet packets, which generate a basis of the signal space. The list of wavelet packets, which generate a Best Basis tree is being produced by the function `WP_BB_Listree1D_P.m` from the *W_packetsP* folder.

`D_List_synthP_2D.m` Does the same for two-dimensional signals. The list of 2D wavelet packets, which generate a Best Basis tree is being produced by the function `WP_BB_Listree2_D_P.m` from the *W_packetsP*.

`D_wP_coefP.m` This function computes the DFT spectra of discrete spline wavelet packets of various orders.

Discr_SplineRMPursuitP

This folder is associated with Sect. 14.5.

`RMP_runD.m` This code launches the interactive process of the Regularized Matching Pursuit of columns of an image, which are blurred by a Gaussian kernel, and/or are corrupted by additive noise.

`loadParametersMP_D.m` This function supports the user interface for setting the processing parameters.

`buildInputFunctions_RMP_D.m` This function prepares the orinal image, the blurred image, noised blurred input, the DFT of the blurring kernel and an additive noise.

`D_wP_theta_coefP.m` This function calculates the DFT of the discrete spline wavelet packets for the approximating dictionary and the DFT of testing waveforms (wavelet packets convolved with the kernel).

`D_nois_model.m` This function emulates the additive noise by analysis of the input signals, as is described in Sect. 10.2.2.

`D_purs_optP.m` This function computes the oblique projection coefficients of the remainder. In addition, it is finding an optimal waveform, and determines its position.

`D_rm_pursP.m` This function implements the Regularized Matching Pursuit of a row signal. It does the main job in the folder.

`D_RMP_displayP.m` This function displays the result of the Regularized Matching Pursuit.

B_waveletsP

This folder is associated with Chap. 16.

`BW_anal.m` This function implements the forward lifting biorthogonal wavelet transform of a row signal or an array of row signals, using prediction filters derived from polynomial and discrete splines.

`BW_anal_2D.m` Does the same for two-dimensional signals.

`down_lif.m` This function implements one step of the direct biorthogonal wavelet transform.

`slfilt_lif.m` This function provides spline-based prediction filters for lifting biorthogonal transforms.

`BW_synth.m` This function implements the inverse lifting biorthogonal wavelet transform. It restores a row signal or an array of row signals from the set of the wavelet transform coefficients.

`BW_synth_2D.m` Does the same for two-dimensional signals.

FramesP

This folder is associated with Chap. 17.

`hmwf.m` This function prepares the set of filters for a given frame transform, using the functions `FW3_P.m` and `HM4_P.m`.

`FW3_P.m` This function prepares the set of filters for one step of a given frame transform, which is implemented by a three-channel filter bank.

`HM4_P.m` This function prepares the set of filters for for one step of a given frame transform, which is implemented by a four-channel filter bank.

`fram34_dec_hm.m` This function provides the frame decomposition of a signal or an array of row signals. The frame transforms are implemented by either three-channel or four-channel filter banks, involving either `FW3_P.m` or `HM4_P.m` functions, respectively.

`fram34_dec2d_hm.m` Does the same for two-dimensional signals.

`fram34_dec_hmwf.m` This function provides the frame decomposition of a signal or an array of row signals. The frame transforms are implemented by either three-channel or four-channel filter banks, using filters prepared by the function `hmwf.m`. It is advisable when the frame transforms are applied repeatedly as, for example, in the Bregman Iterations.

`fram34_dec2d_hmwf.m` Does the same for two-dimensional signals.

`fram34_rec_hm.m` This function implements synthesis of a row signal or an array of row signals from the set of the frame transform coefficients. The inverse frame transforms are implemented by either three-channel or four-channel filter banks, involving either `FW3_P.m` or `HM4_P.m` functions, respectively.

`fram34_dec2d_hm.m` Does the same for two-dimensional signals.

`fram34_rec_hmwf.m` This function implements synthesis of a row signal or an array of row signals from the set of the frame transform coefficients. The inverse frame transforms are implemented by either three-channel or four-channel filter banks, using filters prepared by the function `hmwf.m`. It is advisable when the frame transforms are applied repeatedly as, for example, in the Bregman Iterations.

`fram34_dec2d_hmwf.m` Does the same for two-dimensional signals.

`fram3_down_hm.m` This function implements a single step of the three-channel frame decomposition of a signal or an array of row signals, involving the function `FW3_P.m`.

`fram4_down_hm.m` This function implements a single step of the three-channel frame decomposition of a signal or an array of row signals, involving the function `HM4_P.m`.

`fram3_down_hmwf.m` This function implements a single step of the three-channel frame decomposition of a signal or an array of row signals, using filters prepared by the function `hmwf.m`.

`fram4_down_hmwf.m` This function implements a single step of the three-channel frame decomposition of a signal or an array of row signals, using filters prepared by the function `hmwf.m`.

`fram3_up_hm.m` This function implements a single step of the three-channel frame reconstruction of a signal or an array of row signals, involving the function `FW3_P.m`.

`fram4_up_hm.m` This function implements a single step of the three-channel frame reconstruction of a signal or an array of row signals, involving the function `HM4_P.m`.

`fram3_up_hmwf.m` This function implements a single step of the three-channel frame reconstruction of a signal or an array of row signals, using filters prepared by the function `hmwf.m`.

`fram4_up_hmwf.m` This function implements a single step of the three-channel frame reconstruction of a signal or an array of row signals, using filters prepared by the function `hmwf.m`.

`list_frame3.m` This script comprises the list of the available frames generated by spline-based three-channel filter banks.

`list_frame4.m` This script comprises the list of the available frames generated by spline-based four-channel filter banks.

`Lis_fram.m` This code comprises the list of indices of available frames generated by spline-based filter banks. To be used by the function `compare_frames.m` from the folder *Bregman_iteration*.

Bregman_iteration

This folder is associated with Chap. 18.

`run_Bregman_1d.m` This code launches the interactive process of the restoration of columns of an image by 1D Bregman iterations. The columns are blurred by a Gaussian kernel, and/or are corrupted by additive noise and a number of samples are missing.

`loadParameters_Breg_1d.m` This function supports the user interface for setting the processing parameters.

`BREGMAN_INPAINT_DEBLU_hm_1d.m` This function implements a single Bregman iteration for the restoration of a row signal.

`framinPb_1d.m` This function implements linear operations with one-layer cell arrays.

`conjgradBP_1d.m` 1D conjugate gradient solver.

`fram_threPb_1d.m` This function implements soft thresholding of a one-layer cell array of the frame transform coefficients.

`run_Bregman.m` This code launches the interactive process of the restoration of an image by 2D Bregman iterations. The image is blurred by either a Gaussian or a motion kernel, and/or is corrupted by additive noise and a number of pixels are missing.

`loadParameters_Breg.m` This function supports the user interface for setting the processing parameters.

`BREGMAN_INPAINT_DEBLU_hm.m` This function implements a single Bregman iteration for the restoration of an image.
`framinPb.m` This function implements linear operations with cell arrays.
`conjgradBP.m` 2D conjugate gradient solver.
`fram_threPb.m` This function implements soft thresholding of a cell array of the frame transform coefficients.
`display_breg.m` This code displays the result of the Bregman iterations.
`compare_frames.m` This code compares performance of different frames on restoration of a corrupted image. The PSNR results are saved in the file `TAB.txt`.
`loadParameters_Breg.m` This function supports the user interface for seting the processing parameters for the frames comparison.

Utilities

This folder contains auxiliary codes.

`betw.m` This function inserts columns of an array inbetween columns of another array.
`circ_h.m` Circular shift of a row signal or an array of row signals.
`rshuk.m` Right shift of of a row signal or an array of row signals..
`lshuk.m` Left shift of of a row signal or an array of row signals..
`ushuk.m` Up shift of of a column signal or an array of column signals..
`dshuk.m` Down shift of of a column signal or an array of column signals..
`evod.m` Separation of even and odd columns of an array.
`exteB.m` Symmetric extension of an array.
`shrinB.m` Shrinkage of an array.
`havshB.m` Split an array into two halves in horizontal direction.
`havsvB.m` Split an array into two halves in vertical direction.
`quartB.m` Split a square array into four squares.
`surqua.m` This function inserts a sub-array into upper left corner of a bigger array.

Store

This folder contains a few image files in `.mat`, `.bmp`, `.tiff` and `.jpg` formats, which are used in applications.

Glossary

$\mathbb{Z}$ and $\mathbb{N}$ sets of integer and natural numbers, respectively

$\cdot^*$ complex conjugate

$\binom{n}{k}$ the binomial coefficient

$\mathscr{P}_T$ space of T-periodic continuous-time signals

$\mathscr{P}^2_{X,Y}$ space of X, Y-periodic 2D continuous-time signals

C^p space of functions, which have $p2$ continuous derivatives

$c_n(f)$ the Fourier coefficient of a periodic signal $f(t)$

$\Pi[N]$ space of N-periodic discrete-time signals

$\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$, where $N = 2^j$, $j \in \mathbb{N}$

$\delta[n]$ Kronecker delta

$\tilde{\delta}[k] \stackrel{\text{def}}{=} \delta[k](\text{mod}N)$ the N-periodized Kronecker delta

$\Pi[N]$ space of N-periodic discrete-time signals

$\hat{x}[n] = \sum_{k=0}^{N-1} x[k]\,\omega^{-nk}$ DFT of signal $\mathbf{x} \in \Pi[N]$

$\hat{x}[n]_m = \sum_{k=0}^{N/2^m-1} x[k]\,\omega^{-2^m nk}$ DFT of signal $\mathbf{x} \in \Pi[N/2^m]$

p-filter digital periodic filter

$\mathbf{x}_0$ **and** $\mathbf{x}_1$ the even and the odd polyphase components of a signal $\mathbf{x}$

$\tilde{Z}[f]_n(t)$ periodic Zak transform

$\chi[a, b](t)$ characteristic function of the interval (a, b) (Eq. (3.12))

$B^p(t)$ N-periodic B-spline of order p

${}^p\mathscr{S}_T$ space of T-periodic splines of order p

${}^p\mathscr{S}$ space of N-periodic splines of order p on the grid $\{k\}$, $k \in \mathbb{Z}$

${}^p\mathscr{S}_m$ space of $N-$periodic splines of order p on the grid $\{2^m k\}$, $k \in \mathbb{Z}$

$\zeta^p[n](t)$ periodic exponential splines of order p on the grid $\{k\}$, $k \in \mathbb{Z}$

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4,

$u^p[n] \stackrel{\text{def}}{=} \zeta^p[n](0)$ characteristic sequence of periodic spline space ${}^p\mathcal{S}$

$\gamma^p[n](t) = \zeta^p[n](t)/\sqrt{N\,u^{2p}[n]}$ normalized periodic exponential splines of order p

$N_m \stackrel{\text{def}}{=} N/2^m = 2^{j-m}$

$\bar{N}_m \stackrel{\text{def}}{=} N/3^m$

$V^p[n] \stackrel{\text{def}}{=} \frac{u^p[n]}{\sqrt{u^{2p}[n]}}$ Eq. (4.20)

$R^{p,q}[n] \stackrel{\text{def}}{=} \sqrt{\frac{u^{2(p+q)}[n]}{u^{2p}[n]\,u^{2q}[n]}}$ Eq. (4.21)

$W^{p,s}[n] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi n}{N}\right)^{2s} \frac{u^{2(p-s)}[n]}{u^{2p}[n]}$ Eq. (4.19)

$Q^{p,q}[n] \stackrel{\text{def}}{=} \frac{u^{p+q}[n]}{\sqrt{u^{2p}[n]\,u^{2q}[n]}}$ Eq. (4.33)

$\hat{x}[n]_{\bar{m}} \stackrel{\text{def}}{=} \sum_{k=0}^{\bar{N}_m-1} e^{-2\pi i n k/\bar{N}_m} x[k]$ "Triadic" DFT of an $\bar{N}_m-$periodic signal $\mathbf{x} = \{x[k]\}$, where $\bar{N}_m \stackrel{\text{def}}{=} N/3^m$

$w^{2r}[n] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi n}{N}\right)^{2r}$ Eq. (5.3)

$W^r[\kappa,\nu] \stackrel{\text{def}}{=} \left(2\,\sin\frac{\pi\kappa}{N}\right)^{2r} u^{2r}[\kappa]\,u^{4r}[\nu]$ (Eq. (5.12))

$\zeta^p[\kappa,\nu](x,y) \stackrel{\text{def}}{=} \zeta^p[\kappa](x)\,\zeta^p[\nu](y)$

$u^p[\kappa,\nu] \stackrel{\text{def}}{=} u^p[\kappa]\,u^p[\nu]$

$PSNR \stackrel{\text{def}}{=} 10\log_{10}\left(\frac{M\,255^2}{\sum_{k=1}^{M}(x_k-\tilde{x}_k)^2}\right)\,dB$ Eq. (6.42)

$B_m^p(t)$ The N-periodic normalized B-spline in the space ${}^p\mathcal{S}_m$ (Eq. (8.2))

$\zeta_m^p(t)$ The N-periodic exponential spline in the space ${}^p\mathcal{S}_m$ (Eq. (8.4))

$u_m^p[n]$ The N_m-periodic characteristic sequence of the space ${}^p\mathcal{S}_m$ (Eq. (8.5))

$\gamma_m^p(t)$ The N-periodic normalized exponential spline in the space ${}^p\mathcal{S}_m$ (Eq. (8.9))

$s_{m-1}[n] \stackrel{\text{def}}{=} \sqrt{\frac{u_{m-1}^{4r}[n]}{2\,u_m^{4r}[n]}}\,\cos^{2r}\frac{\pi n}{N_{m-1}}$ (Eq. (8.14))

$d_{m-1}[n] \stackrel{\text{def}}{=} \omega^{2^{m-1}n}\,s_{m-1}[n+N_m] = \omega^{2^{m-1}n}\sqrt{\frac{u_{m-1}^{4r}[n+N_m]}{2\,u_m^{4r}[n]}}\,\sin^{2r}\frac{\pi n}{N_{m-1}}$ (Eq. (8.16))

$\psi_{m,l}^{2r}(t)$ spline wavelet packet of order $2r$ (Eq. (8.46))

$\Psi_{m,l}^{2r}$ discrete-time wavelet packet of order $2r$ (Eq. (9.18)) in space $\Pi[N]$

$\Psi_{m,l,\tilde{l}}^{2r}[k,n] \stackrel{\text{def}}{=} \Psi_{m,l}^{2r}[k]\,\Psi_{m,\tilde{l}}^{2r}[n]$ 2D discrete-time wavelet packet of order $2r$ in space $\Pi[N,N]$

$T^{2r,q}[\iota] \stackrel{\text{def}}{=} \frac{u^{2r+q}[\iota]}{\sqrt{u^{4r}[\iota]\,u^{q}[\iota]}}$ Eq. (11.26)

$\mathbf{b}^{2r}_{[m]} = \left\{ b^{2r}_{[m]}[k] \right\},\ k \in \mathbb{Z}$, centered N-periodic discrete B-spline of order $p = 2r$ of span $K = 2^m$ (Sect. 13.1.1)

${}^{2r}\mathscr{S}_{[m]}$ the space of the N-periodic discrete splines of span $K = 2^m$

$\zeta^{2r}_{[m]}(\nu) = \left\{ \zeta^{2r}_{[m]}(\nu)[k] \right\}$ discrete periodic exponential spline in the space ${}^{2r}\mathscr{S}_{[m]}$ (Eq. (13.5))

$U^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \zeta^{2r}_{[m]}(\nu)[0]$ characteristic sequence of the space ${}^{2r}\mathscr{S}_{[m]}$ (Eq. (13.12))

$\gamma^{2r}_{[m]}(\nu) = \left\{ \zeta^{2r}_{[m]}(\nu)[k] \right\}$ normalized discrete periodic exponential spline in the space ${}^{2r}\mathscr{S}_{[m]}$ (Eq. (13.18))

$\breve{W}^{r,d}_{[m]}[\nu] srr \left(\frac{2\sin(2^m \pi\ \nu/N)}{2^m} \right)^{4d} \frac{U^{4(r-d)}_{[m]}[\nu]}{U^{4r}_{[m]}[\nu]}$ (Eq. (13.22))

$V^{2r}_{[m]}[\nu] \stackrel{\text{def}}{=} \frac{U^{2r}_{[m]}[\nu]}{\sqrt{U^{4r}_{[m]}[\nu]}}$ (Eq. (13.23))

$\overline{\hat{x}}[\nu] \stackrel{\text{def}}{=} \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x[k]\, \omega^{-k\nu}$ normalized DFT

$x_+ \stackrel{\text{def}}{=} (x + |x|)/2$

${}^{2r}\mathscr{W}_{[m],l}$ the space of the N-periodic discrete splines wavelet packets of span $K = 2^m$

$\gamma^{2r}_{[m],l}(\nu)$ normalized discrete periodic exponential spline in the space ${}^{2r}\mathscr{W}_{[m],l}$ (Eqs. (14.8) and (14.9))

$\psi^{2r}_{[m],l}$ discrete spline wavelet packets from the subspace ${}^{2r}\mathscr{W}_{[m],l}$ (Eq. (14.30))

Index

A. Z. Averbuch et al., *Spline and Spline Wavelet Methods with Applications to Signal and Image Processing*, DOI: 10.1007/978-94-017-8926-4,