Chapter 2
Correlation

Correlation is a function in the time-delay domain that represents a possible relationship between two functions. A signal analysis technique that uses the correlation function is referred to as "correlation analysis," a technique that was commonly used for signal analysis until the 1960s. But correlation analysis has been pushed to a backseat role since the appearance of the FFT technique. This is because the cross- (and auto-) correlation and cross- (and auto-) power spectra forms a Fourier transform pair, and the correlation functions are calculated much faster from the cross- and auto-spectra by using the Fast Fourier Transform.

However, understanding of the correlation function is very important as the basis of signal analysis, even in an era when the DFT is the main tool of modern signal analysis.

2.1 Similarity of Two Sequences of Numbers

To make the meaning of the correlation clear, let us consider the similarity between two sequences of numbers, $x(n)$ and $y(n)$. This will in due course lead to the derivation of a formula for the cross-correlation function.

The sequences $x(n)$ and $y(n)$ need not be time signals, however, since the primary interest is in time signal analysis, we assume that they are samples of time signals. They can be either real or artificial. The number n represents the discrete time and therefore it is treated as the time itself.

If $x(n)$ and $y(n)$ are identical, the similarity is the greatest. In order to quantify the degree of similarity it will suffice to calculate squares of differences of the two functions at corresponding times and add them up.

Even if the two waveforms are the same but the magnitudes are different, there are differences between them. If the numerical values are large, the sum of the squares of the difference will be large and vice versa. Therefore, a normalization procedure is necessary so that the result is not affected by the magnitudes of the waveforms.

K. Kido, *Digital Fourier Analysis: Advanced Techniques*,
DOI: 10.1007/978-1-4939-1127-1_2,
© Springer Science+Business Media New York 2015

For the normalization, the sequences should be divided by their respective standard deviations. The standard deviation of a sequence is actually its effective amplitude, i.e., the root-mean-square of the power of the alternating component. The division by the standard deviations removes the differences of magnitudes of the waveforms. We are more interested in the shapes of the waveforms.

If the signals contain constants (i.e., dc components), they can be removed and the similarity of the "changes of the waveforms" can be discussed.

Consider two sequences with zero means and normalized by respective standard deviations. The difference between the two sequences can be represented by the sum of the squares of the differences of values at corresponding times. If one of the sequence is shifted on the time axis, the difference (or, conversely, similarity) of the two sequences will change. This dependence of the similarity on the amount of the time shift is important information. This time shift is referred to as (*time*) *lag*. The similarity is therefore a function of the time lag.

In order to normalize the sequences, $x(n)$ and $y(n)$, their standard deviations, σ_x and σ_y, must be calculated. The standard deviations with sequence length L are defined by Eqs. (2.1) and (2.2).

$$\sigma_x = \sqrt{\frac{1}{L}\sum_{n=0}^{L-1}\{x(n)-\bar{x}\}^2} \tag{2.1}$$

$$\sigma_y = \sqrt{\frac{1}{L}\sum_{n=0}^{L-1}\{y(n)-\bar{y}\}^2} \tag{2.2}$$

where $\bar{x}$, $\bar{y}$ are means of $\{x(n)\}$ and $\{y(n)\}$, respectively. At present, it is assumed that the means are zero for simplicity.

The sum of squares of the differences of the sequence $x(n)$ and m-time- shifted sequence $y(m+n)$, both normalized by the individual standard deviations, is given by

$$e_m^2 = \frac{1}{L}\sum_{n=0}^{L-1}\left(\frac{x(n)}{\sigma_x}-\frac{y(m+n)}{\sigma_y}\right)^2 \tag{2.3}$$

This is modified as

$$\begin{aligned} e_m^2 &= \frac{1}{L}\sum_{n=0}^{L-1}\left(\frac{x(n)^2}{\sigma_x^2}+\frac{y(m+n)^2}{\sigma_y^2}-2\frac{x(n)y(m+n)}{\sigma_x\sigma_y}\right) \\ &= 2\left(1-\frac{1}{L\sigma_x\sigma_y}\sum_{n=0}^{L-1}x(n)y(m+n)\right) \end{aligned} \tag{2.4}$$

The value of this equation is equal to 0 when normalized $x(n)$ and $y(m + n)$ are identical. Therefore, it better satisfies our intention to calculate the inverse degree of "similarity" of the two sequences.

2.2 Cross-Correlation Function

Equation (2.4) represents the "degree of similarity" between the two functions. Considering only the case, $m = 0$, if the two functions are identical, it is equal to 0; if there is no relation between them, it is equal to 2; and if they have the same magnitudes and the signs are opposite, it is equal to 4. Thus, it is not necessary to use the whole of Eq. (2.4). The second term may be more suitable since it becomes 1 when the two sequences are identical, 0 when they are random, and -1 when they have the identical magnitude but opposite signs.

The second term of Eq. (2.4) can be used to represent the degree of similarity of the two functions.

$$r_{xy}(m) = \frac{1}{L\sigma_x\sigma_y}\sum_{n=0}^{L-1} x(n)y(n+m) \tag{2.5}$$

$$= \frac{\sum_{n=0}^{L-1} x(n)y(n+m)}{\sqrt{\sum_{n=0}^{N-1} x^2(n)}\sqrt{\sum_{n=0}^{N-1} y^2(n)}} \tag{2.6}$$

This is referred to as the *cross-correlation function*. As the above equations show, the correlation function is a function of lag m.

The convolution equations, Eqs. (1.1) and (1.2), and the cross-correlation equations, Eqs. (2.5) and (2.6), are almost the same. The difference is that convolution is a function of n obtained by adding terms $x(n-p)\ h(p)$ for all p, while correlation is a function of m obtained by adding terms $x(n)\ y(m + n)$ for all n. This is illustrated in Fig. 2.1.

First, consider the convolution function. The output of a system at a discrete time n is the addition of the products of

(1) $x(n)$, the input at time n, and the impulse response $h(0)$,
(2) $x(n-1)$, the input at one sample time before n, and the impulse response $h(1)$,
…
(3) $x(n-p)$, the input at p sample time before n, and the impulse response $h(p)$,
…

Figure 2.1a illustrates this process. The time-reversed impulse response $h(k)$ is displayed so that $k = 0$ corresponds to the discrete time n. Then, the response of

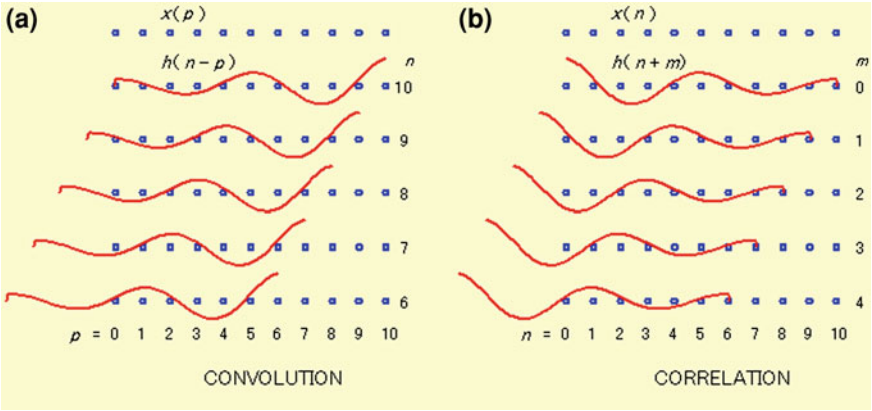


Fig. 2.1 Comparison of handling of data between the convolution and the cross-correlation calculations

the system $y(n)$ is obtained by adding the products of each of $x(p)$ (shown by the top row) and its corresponding term $h(n\text{-}p)$ (shown by the second row for $n = 10$, the third row for $n = 9$, …).

In contrast, as Fig. 2.1b shows, the cross-correlation between $x(n)$ and $h(n)$ is obtained by multiplying each $x(n)$ with its corresponding $h(n)$ with the latter forward-shifted by m samples (shown by the $(m-1)$-th row in (b)) and adding all the product terms for each m.

As Fig. 2.1 shows, the convolution of $x(n)$ and $h(n)$ can be obtained as the cross-correlation of the two functions when one of the two functions is time reversed.

In Eq. (2.6), the correlation operation is defined between two sequences with the finite length L. If this number is small, correlation shows a property of limited portions of the two sequences. In the derivation of Eq. (2.5), continuous data of length L are used from $n = 0$ to $L-1$. In this case, however, the length L must be very large to represent the whole length of the sequences. Instead of using the continuous data as in Eq. (2.5), the cross-correlation can be defined in the form of an ensemble mean shown below.[1]

$$r_{xy}(m) = \frac{1}{\sigma_x \sigma_y} < x_n y_{n+m} > \tag{2.7}$$

[1] If the number of observed data is large enough to represent the property of the parent population, separations among data do not have to satisfy the sampling theorem. However, m must satisfy the criterion of the sampling theorem.

From the standpoint of signal analysis, the correlation of signals in a short period that satisfies a specific condition is sometimes more useful. It is also more convenient if the length L is finite for the practical reason of calculation. If analysis of a long time range is necessary, the equation below is useful, which makes it possible to keep the number of data equal to L while extending the effective analysis length equal to PL.

$$r_{xy}(m) = \frac{1}{L\sigma_x\sigma_y}\sum_{n=0}^{L-1} x(nP)y(nP+m) \tag{2.8}$$

n in Eq. (2.5) is multiplied by P, which means analysis range L is P-times extended, but the correlation function is correctly calculated so long as the sampling period is unchanged. There is a chance that the result is affected by the properties of contaminating noises related to a specific value of P. If this is the case, the value of P should be randomly varied.

The reader should understand from the derivation of the cross-correlation function that, if $y(n)$ is generated by giving lag m to $x(n)$, the cross-correlation between them will be 1 when the lag is equal to m. Figure 2.2 shows the case with random signal $x(n)$.

The waveforms of the top two rows in Fig. 2.2 are $x(n)$ and $y(n)$ are computer simulated. The signal $y(n)$ is made from $x(n)$ by giving a 25-sample delay. It is difficult to see that they are actually the same signal until one is informed. The cross-correlations calculated by using Eq. (2.5) are shown by the bottom three rows of the figure for $L = 10$, 100, and 1,000. The reader can barely see a peak at $m = 25$ for $L = 10$. For $L = 100$, one can almost say that there is only one peak at $m = 25$. If L is made larger up to 1,000, the noisy up-downs almost disappear. Even for an ideal random sequence simulated by a computer, a large number of averages is necessary. This suggests that averaging must be quite large when the correlation is applied to actual data.

Figure 2.2 shows the results when Eq. (2.5) is directly applied to the data. The results show that the averaging of 10 sequences is too small in order to detect the correlation. Equation (2.8) may help extend the range of averaging.

The above example is a case with no external noise. Even if a noise is added to the data, the effect of the noise contamination will be reduced by increasing the number of averaging. Figure 2.3 shows a case when a random noise with nearly the same power but with no correlation with $y(n)$ is added to $y(n)$. For $L = 10$, it became more difficult to identify the peak at $m = 25$ than the previous example. However, with increased averaging number, the position of the peak is made clear.

As shown above, if two sequences contain common signals with some delay between them, the existence of a common component and the delay time can be found by use of the cross-correlation function.

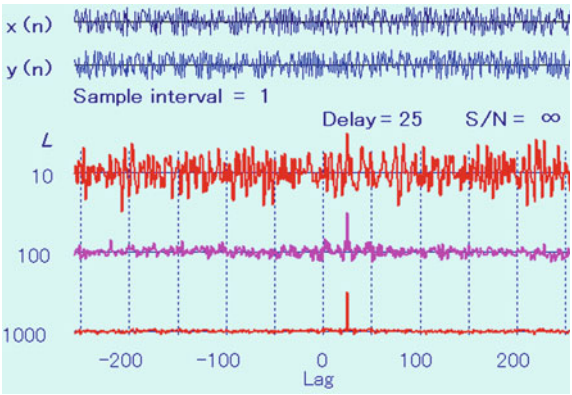


Fig. 2.2 Comparison of the cross-correlation functions of a random sequence and its 25-sample delayed sequence for averaging numbers $L = 10$, 100, and 1,000. Animation available in supplementary files under filename E9-02_Correlation.exe

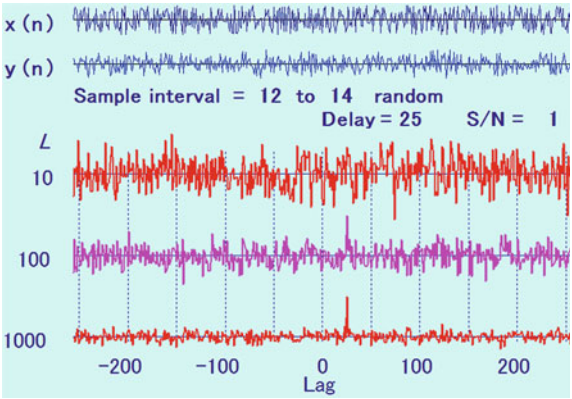


Fig. 2.3 Cross-correlation functions when a random noise with the same power but with no correlation with $x(n)$ is added to $y(n)$. The sequence $x(n)$ and $y(n)$ before the noise addition are the same as those in Fig. 2.2. Animation available in supplementary files under filename E9-03_Correlation.exe

2.3 Cross-Correlation Function Between Input and Output Signals of a Transfer System

The output $y(n)$ of a transfer system with impulse response $h(n)$ and input $x(n)$ is given by a convolution between $x(n)$ and $h(n)$. Since $y(n)$ is made from $x(n)$, there should be some kind of relationship between $x(n)$ and $y(n)$, and it should appear in the cross-correlation.

By taking the range of p from 0 to $K-1$ in Eq. (1.1), the convolution between $x(n)$ and $h(n)$ is given by

$$y(n) = \sum_{p=0}^{K-1} x(n-p)h(p) \tag{2.9}$$

Substituting this equation into Eq. (2.5), the cross correlation r_{xy} is obtained

$$r_{xy}(m) = \frac{1}{L\sigma_x\sigma_y}\sum_{n=0}^{L-1} x(n)\sum_{p=0}^{K-1} x(n+m-p)h(p)$$

Reversing the order of summation,

$$r_{xy}(m) = \frac{1}{L\sigma_x\sigma_y}\sum_{p=0}^{K-1} h(p)\sum_{n=0}^{L-1} x(n)x(n+m-p) \tag{2.10}$$

is obtained. If $x(n)$ is a random sequence, $x(n)x(n+m-p)$ takes positive and negative random values except for the case $m = p$. Therefore, if L is large enough, $x(n)x(n+m-p)$ becomes negligibly small compared to the value for $m = p$,

$$\sum_{n=0}^{L-1} x(n)x(n+m-p) = \sum_{n=0}^{L-1} x(n)x(n) = L\sigma_x^2$$

Then, if $x(n)$ is a random sequence, Eq. (2.10) is given by

$$r_{xy}(m) = \frac{L\sigma_x^2}{L\sigma_x\sigma_y}h(m) = \frac{\sigma_x}{\sigma_y}h(m) \tag{2.11}$$

Equation (2.11) shows the interesting result that the correlation is proportional to the impulse response. The effect of noise contamination is not considered in the above discussion. Even if uncorrelated noise is added to the input and/or output, the same equation Eq. (2.11) is obtained. Its derivation is given in Appendix 2A.

Let us check by numerical calculations whether the cross-correlation gives the impulse response or not.

The impulse response $h(n)$ used here is a decaying sine wave, which was also used in Chap. 1. Since $x(n)$ must be a random sequence, a random number sequence simulated by computer is used. The sequence $y(n)$ is given by the convolution of $x(n)$ and $h(n)$ and then the correlation is calculated between $x(n)$ and $y(n)$. The number of averages is examined for three cases, 10, 100, and 1,000.

The results are given in Fig. 2.4. The three charts from the top show the impulse response $h(n)$, random sequence $x(n)$, and the system output $y(n)$, respectively. The bottom three charts show the cross-correlation functions for $L = 10$, 100, and 1,000, respectively. The spacing of data (P in Eq. (2.8)) is randomly varied within 12 and 14. The results show that the averaging of 100 is not enough, but if it is increased to 1,000, a waveform similar to $h(n)$ appears in y(n) from around lag $m = 50$, which was given to $y(n)$. As the number of averages is increased, the cross-correlation function gets closer to the true impulse response, indicating that Eq. (2.11) is valid.

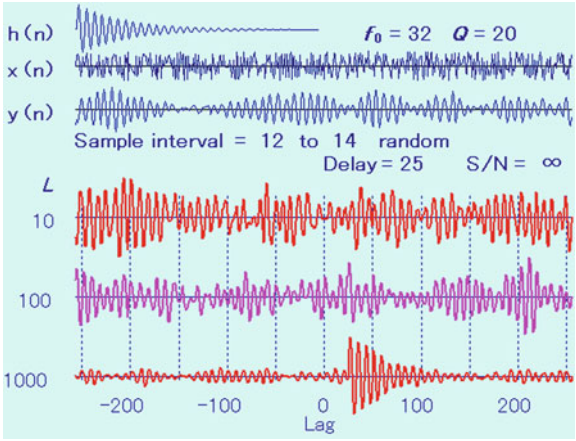


Fig. 2.4 Cross-correlations between a random input $x(n)$ and an output $y(n)$ of a transfer system with an impulse response $h(n)$ for different averaging numbers, 10, 100, and 1,000. Animation available in supplementary files under filename E9-04_Correlation.exe

More insight into cross correlation analysis can be obtained by examining the program for the data given in Fig. 2.4. It will be seen for example that the addition of noise does not deteriorate the results too much, and that a choice of data spacing P with respect to the period of $x(n)$ sometimes gives serious problems, that are not described in the text.

2.4 Auto-Correlation Function

Up to the present, relationship between two different sequences $x(n)$ and $y(n)$ has been considered. During the discussion, the special case $x(n) = y(n)$ was not excluded. In fact, if this is the case, the correlation may include some information about the sequence itself. The *auto-correlation function* is defined by replacing $y(n)$ by $x(n)$ in Eqs. (2.5) and (2.7).

$$r_{xx}(m) = \frac{1}{L\sigma_x^2} \sum_{n=0}^{L-1} x(n)x(m+n) \tag{2.12}$$

$$r_{xx}(m) = \frac{1}{\sigma_x^2} < x(n)x(m+n) > \tag{2.13}$$

It is clear that the auto-correlation function is always equal to 1 for $m = 0$. If the range of n is infinite, the two summations $\sum_n x(n)x(m+n)$ and $\sum_n x(n-m)x(n)$, i.e., $\sum_n x(n)x(-m+n)$ become equal, indicating that auto-correlation is an even function of the lag m.

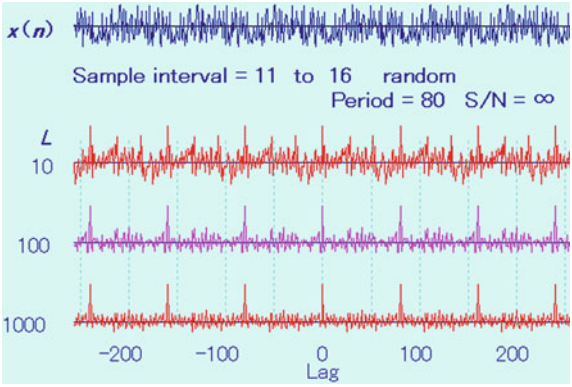


Fig. 2.5 Random sequence with period 80, $x(n)$, and its auto-correlations for averaging numbers $L = 10$, 100, 1,000. Animation available in supplementary files under filename E9-05_AutoCorrelation.exe

If the sequence is random, the auto-correlation must be zero except for $m = 0$. Random number generation functions installed in computers are almost perfect and the cross-correlation function of a random sequence generated by a computer almost always satisfies the condition that it is equal to 1 for $m = 0$ and 0 for $m \neq 0$.

Let us calculate a cross-correlation function of a periodically repeated random sequence generated by a computer. Results of the calculation are shown in Fig. 2.5 for averaging numbers $L = 10$, 100, and 1,000. Since the sequence is periodic, the auto-correlation function becomes 1 at every one period (80 points in the present case). The spacing of data (P in Eq. (2.8)) is randomly varied between 11 and 16. Figure 2.5 shows that averaging 10 times is not sufficient. It seems necessary that at least 100 averages is necessary when the sequence is random.

The above examples are auto-correlations of artificially generated, i.e., rather ideal, signals. The reader might have the impression that the results are being intentionally idealized by the author. As an example of a real signal, the auto-correlation of the number of sunspots as a function of year will be calculated. The yearly change of the number of the sunspots has been observed for approximately 300 years as shown in Fig. 2.6a. Since the number of sunspots is always positive, the mean of the auto-correlation function takes a non zero value. As a trial, let us directly apply Eq. (2.9) without making the mean equal to zero. The result is given by Fig. 2.6b. Since the auto-correlation is an even function of the lag, only the results for $m \geq 0$ are shown. Since the mean is not zero and other data except for the actually observed 294 data are made equal to zero, the auto-correlation takes larger values as the lag approaches 0 as a general trend. Still, one can read that the period of the sunspot number variation is roughly 11 years.

Figure 2.6c shows a sunspot data, which is modified so that the mean becomes equal to zero and its auto-correlation is shown by Fig. 2.6d. This seems to indicate that the sunspot variation has also a longer period, approximately 10 times that of 11 years. However, the data is too short to purport this theory.

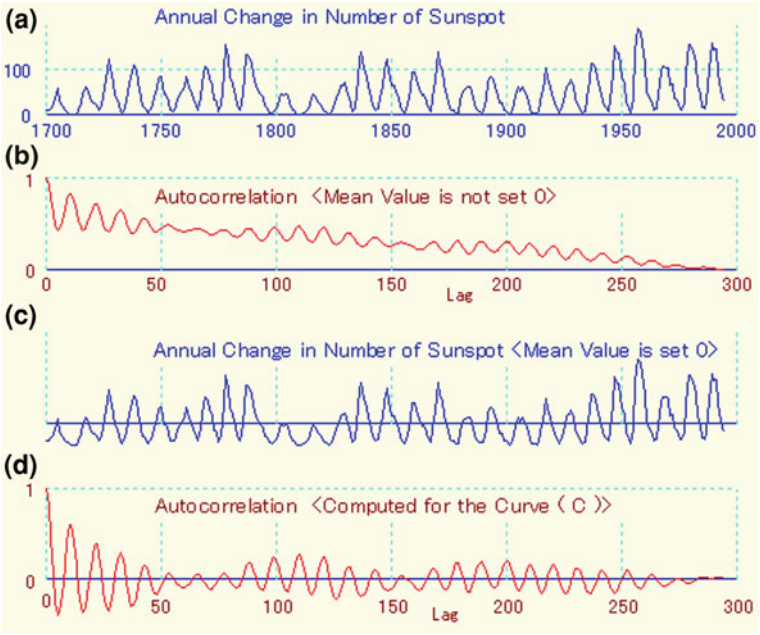


Fig. 2.6 Auto-correlation function of the yearly change of sunspots

2.5 Analysis of Sequences by Auto-Correlation Functions

In Sect. 2.4, applications of extracting periods of sequences were introduced. However, the cross- and auto-correlation functions have wider applications than this.

Let us consider an example shown in Fig. 2.7, which depicts a sound source and a wall. The sound source radiates a random noise. The sound received by the observer is a combination of the direct noise and the noise reflected from the wall. The reflected noise has some time delay compared to the direct sound. Can this time delay be determined by correlation analysis?

The waveforms are represented by sample values. If the direct sound is represented by $x(n)$, the reflected sound can be represented by $rx(n - d)$, where r is the amplitude ratio of the reflected waveform to the direct waveform and d is the time delay between them. Then, the received sound at the observation point $y(n)$ is given by

$$y(n) = x(n) + rx(n - d) \tag{2.14}$$

The auto-correlation of $y(n)$ is given by (see Appendix 2C)

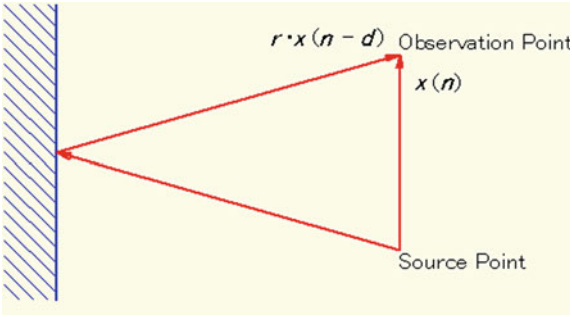


Fig. 2.7 Superposition of the direct sound $x(n)$ and the reflected sound $rx(n-d)$, with r times the amplitude and delay time d

$$\begin{aligned} R_{yy}(m) &= \frac{1}{L\sigma_y^2}\sum_{n=0}^{L-1} y(n)y(n+m) \\ &= (1+r^2)R_{xx}(m) + rR_{xx}(m+d) + rR_{xx}(m-d) \end{aligned} \tag{2.15}$$

Equation (2.15) shows that the correlation function takes the value $(1 + r^2)$ at $m = 0$, and r at $m = \pm d$. Remember that $R_{yy}(0) = 1$ and $R_{yy}(m \neq 0) = 0$ if $x(n)$ is a random sequence.

In order to confirm the above discussion, auto-correlation functions of a random sequence added to the same sequence with a time delay are shown in Fig. 2.8. The time delay corresponds to a time length of 90 samples. There are two peaks at $m = \pm 90$ with the amplitude of 1/2 of the peak at $m = 0$. This is because $r = 1$ in this case and external noise is not included.

Next, consider the auto-correlation of the observed output $y(n)$ of a system with an impulse response $h(n)$ and a random input sequence $x(n)$. This corresponds to the case of cross-correlation function of Fig. 2.4. The results are shown in Fig. 2.9 without a rather lengthy theory.

The first and second charts in Fig. 2.9 shows the output $y(n)$, the convolution of $x(n)$ and $h(n)$, and the impulse response, $h(n)$, respectively. The bottom three charts are auto-correlations for three numbers of averaging, $L = 10$, 100, and 1,000. The data spacing (P in Eq. (2.8)) is randomly chosen from 12 to 20. The impulse response given by the auto-correlation function extends to both positive and negative time directions contrary to the case of the cross-correlation function. This is because the auto correlation is an even function.

Some examples of auto-correlations of waveforms are shown in Fig. 2.10. Some of these will be referred to in later sections.

Figure 2.10a is an auto-correlation of a random sequence. As stated before, it should become 1 at 0 lag and 0 at other lags. However, even with 1,000 averages, it is not perfectly zero for non zero lags.

Figure 2.10b is an auto-correlation of a sequence obtained from a moving-average of the sequence (a) over 64 points ($x'(n) = \frac{1}{64}\sum_{m=-32}^{31} x(n+m)$). The

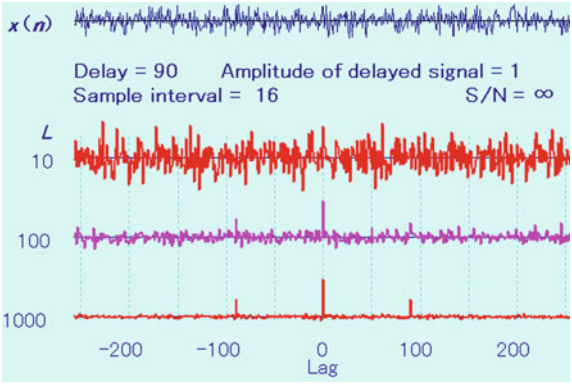


Fig. 2.8 Auto-correlation functions of a random sequence $x(n)$ added with the same sequence with a time delay of $d = 90$. The numbers of averaging L are 10, 100, and 1,000. Animation available in supplementary files under filename E9-08_AutoCorrelation.exe

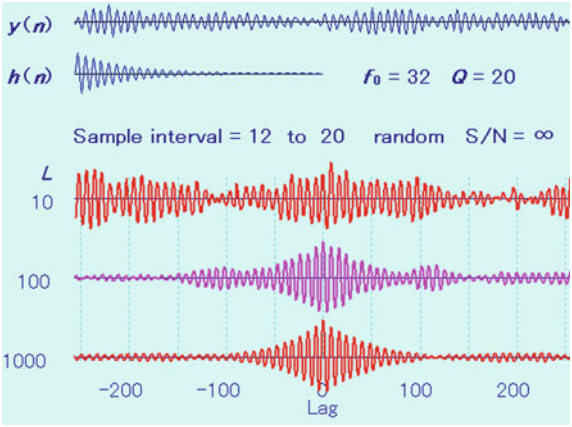


Fig. 2.9 Auto-correlation functions of $y(n)$ as an output of a system with an impulse response $h(n)$ and an input $x(n)$ (not shown). The numbers of averages L are 10, 100, and 1,000. Animation available in supplementary files under filename E9-09_AutoCorrelation.exe

auto-correlation function takes the form of an isosceles triangle with its base equal to twice the time length of the moving average.

Figure 2.10c is an auto-correlation of a sequence obtained similarly as (b) but with Hanning window weighting (see Chap. 7). The auto-correlation function is similar to the shape of the Hanning window.

Figure 2.10d is an auto-correlation of a sequence of positive and negative pulses with a random spacing from 25 to 45. The average of the sequence is equal to zero. The auto-correlation function is zero within $m = \pm 25$ (except for $m = 0$) and, outside of this range, it approaches 0 as the number of averaging increases.

Figure 2.10e is an auto-correlation of a sequence similar to the one in Fig. 2.8, but the original random sequence is a moving-average over 16 points. The auto-correlation function has a spreading similar to Fig. 2.10b.

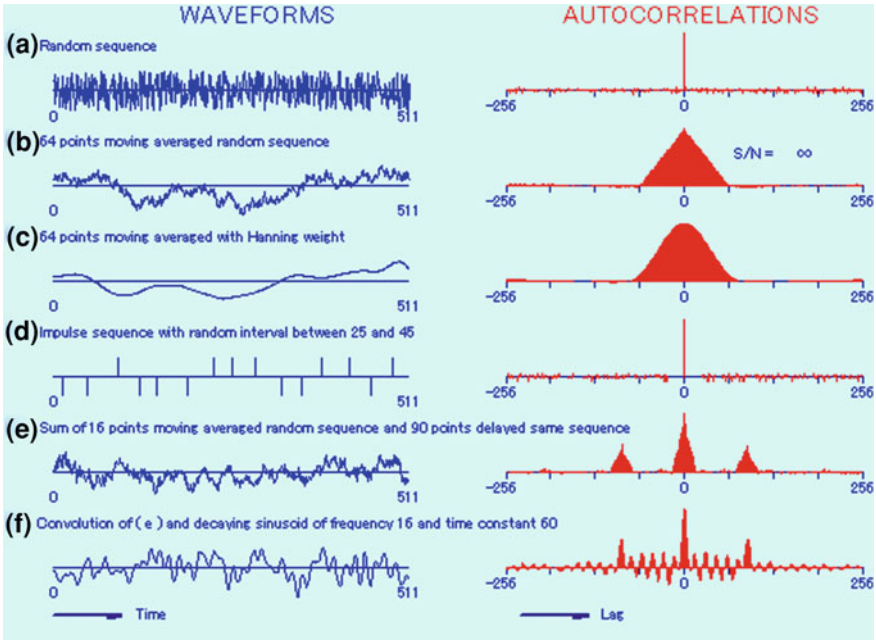


Fig. 2.10 Auto-correlation functions of several waveforms. Animation available in supplementary files under filename E9-10_AutoCorrelation.exe

Figure 2.10f is an auto-correlation of a sequence similar to the one in Fig. 2.9. Since the input sequence is Fig. 2.10e, there are many small triangular peaks around the main three peaks of Fig. 2.10e.

In the program of Fig. 2.10, the reader can check auto-correlations under many other conditions.

In the examples of Figs. 2.8, 2.9 and 2.10, noise is not considered. If an uncorrelated noise is added to the sequences, auto-correlations are decreased except for the lag at $m = 0$. One can also check these aspects in the programs of Figs. 2.8, 2.9 and 2.10.

2.6 Short-Term Auto-Correlation Function

In the previous discussions auto-correlations that represent properties over whole ranges of rather long sequences have been considered. The auto-correlation can be applied to signals that change their waveforms continuously with time, such as speech signals. In those signals, auto-correlations over long ranges of lags are not appropriate. In speech signals, a pitch can last several milliseconds and it can change in several tens to hundreds of milliseconds. The auto-correlation includes

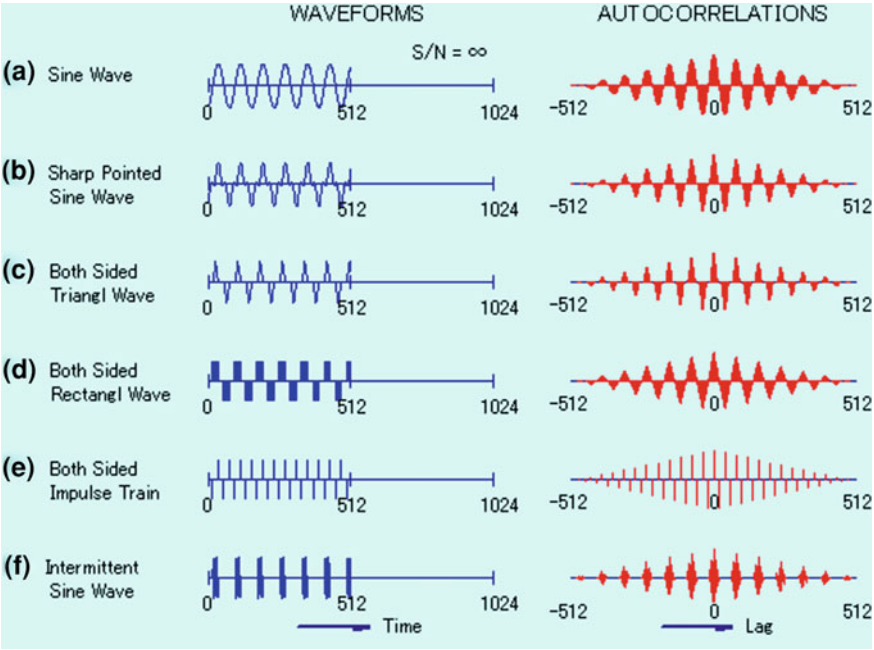


Fig. 2.11 Six periodic waveforms with 512 sequence lengths and their short-term auto-correlation functions with $L = 1024$. Animation available in supplementary files under filename E9-11_Autocorrelation.exe

not only pitch information, but also spectrum information and the latter also changes within a short duration. If the auto-correlation is used for speech analysis, it must be applied to signals with short duration, or to short portions of long signals.

Equation (2.12) can still be applied in those cases where the data length is short and the number of averages (L) cannot be made larger than twice the data length. Let's synthesize several waveforms with short periods and calculate their *short-term auto-correlations*.

Six periodic waveforms and their auto-correlation functions are shown in Fig. 2.11. The sequence length of the signal shown in the figure is 512, and outside of the sequence the signal is considered to be zero. The number of averages L, is 1,024, and therefore, the auto-correlation function reduces linearly from $m = 0$ to $m = \pm 512$. The auto correlation is zero for $|m| > 512$.

Figure 2.11 shows some interesting relations between the waveforms and their auto-correlations functions: (1) the auto-correlations of periodic waveforms also becomes periodic, (2) the auto-correlations of sinusoidal waveforms also become sinusoidal, (3) the auto-correlations of impulse sequences also become impulsive and so on. The periodic intermittent sine wave of Fig. 2.11f is a succession of the

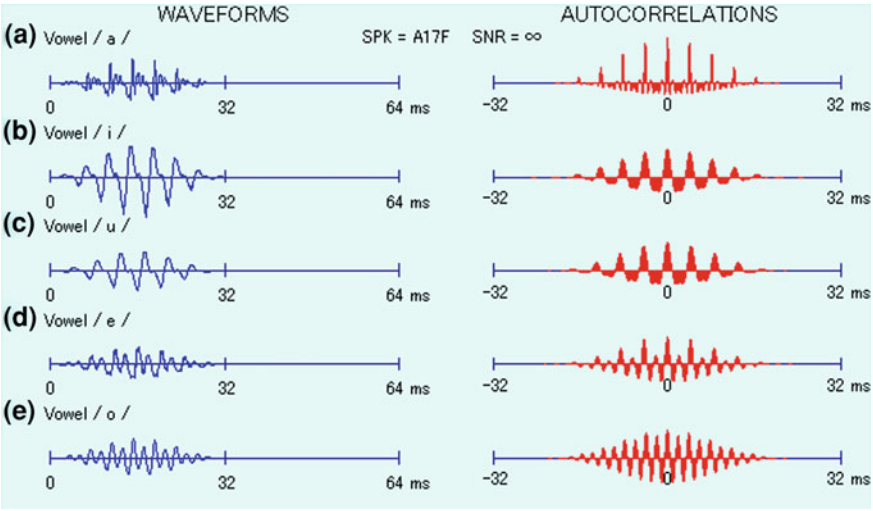


Fig. 2.12 Auto-correlation functions of five vowels with Hanning windows. Animation available in supplementary files under filename E9-12_VowelCorrelation.exe

same short sine waves. Even if each short sine wave is replaced with other types of waves, similar results will be obtained. One can obtain other findings by running the program for Fig. 2.11.

The auto-correlation function is often used to detect pitch frequencies of speech. The left-hand side of Fig. 2.12 shows waveforms of /a/, /i/, /u/, /e/, and /o/ sampled with a 16 kHz sampling frequency through the Hanning window. The first 512 points (32 ms) contain the speech waveforms and 512-point zeros are added to them. The right-hand side shows their auto-correlation functions. Since the waveforms approach zero near the ends due to the Hanning window, the auto-correlation functions quickly reduce as the lag increases in the positive and negative directions.

Since the vowels have periodic waveforms, their auto-correlation functions are also periodic. The largest peak with the minimum lag (except for the zero lag) gives the period of the pitch. They can be easily read from Fig. 2.12a–d, but that of Fig. 2.12e is not. Nevertheless, the largest peak (except that of the origin) gives the period of the pitch.

In the program associated with the analysis of Fig. 2.12, one can see the dependence of the auto-correlations functions of voices on their speakers as well as on the gender of their speakers.

2.7 Auto-Correlations of Sequences of Complex Numbers

So far, all sequences that have been considered are real. The auto-correlation function analysis should be extended to complex numbers for future use. If the sequence in Eq. (2.10) is complex, the auto-correlation is also complex. This can be avoided by changing the sign of the imaginary parts of either of $x(n)$ or $x(m + n)$, as shown below.

$$r_{xx}(m) = \frac{1}{L\sigma_x^2}\sum_{n=0}^{L-1} x^*(n)x(m+n) \tag{2.16}$$

$$r_{xx}(m) = \frac{1}{\sigma_x^2} < x^*(n)x(m+n) > \tag{2.17}$$

where $x^*(n)$ is the complex conjugate of $x(n)$.

The auto-correlation function of a continuous function $x(t)$ is given by Eq. (2.18)

$$r_{xx}(\tau) = \frac{1}{T\sigma_x^2}\int_0^T x^*(t)x(t+\tau)dt \tag{2.18}$$

where T is the range over which the waveform exists. If the signal is infinitely long, the range of integration should be from $-\infty$ to $+\infty$. The standard deviation for the continuous system, σ_x, is given by

$$\sigma_x = \sqrt{\frac{1}{T}\int_0^T x^2(t)dt} \tag{2.19}$$

The cross-correlation is also defined here as

$$r_{xy}(m) = \frac{1}{L\sigma_x\sigma_y}\sum_{n=0}^{L-1} x^*(n)y(n+m) \tag{2.20}$$

$$r_{xy}(m) = \frac{1}{\sigma_x\sigma_y} < x^*(n)y(n+m) > \tag{2.21}$$

The cross-correlation function of continuous functions $x(t)$ and $y(t)$, $r_{xy}(\tau)$, is given by

$$r_{xy}(\tau) = \frac{1}{T\sigma_x\sigma_y}\int_0^T x^*(t)y(t+\tau)dt \tag{2.22}$$

The definitions given above does not cause any change as long as sequences or functions are real. If $x(n)$ and $y(n)$ are complex, the cross-correlation defined by Eqs. (2.20) or (2.21) or (2.22) does not always become real. However, the reader will understand the advantages of the definitions given above when we address the Fourier transforms of auto- and cross-correlation functions.

2.8 Fourier Transforms of Auto-Correlation Functions

Let us calculate the Fourier transform of Eq. (2.18) with the range of integration from $-\infty$ to $+\infty$. It is assumed that $x(t) = 0$ for $|t| > Tx$, where Tx is a sufficiently large positive constant and $x^2(t)$ is integrable. We also neglect the division by T in Eq. (2.18).

$$\mathrm{FT}[r_{xx}(\tau)] = \frac{1}{\sigma_x^2}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^*(t)x(t+\tau)dt\exp(-j2\pi f\tau)d\tau$$

By replacing $(t + \tau)$ by u ($\tau = u - t, \quad d\tau = du$), the above equation is rewritten as

$$\begin{aligned}\mathrm{FT}[r_{xx}(\tau)] &= \frac{1}{\sigma_x^2}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^*(t)x(u)dt\exp\{-j2\pi f(u-t)\}du \\ &= \frac{1}{\sigma_x^2}\int_{-\infty}^{+\infty} x^*(t)\exp(j2\pi ft)dt\int_{-\infty}^{+\infty} x(u)\exp(-j2\pi fu)\}du\end{aligned}$$

Therefore,

$$\mathrm{FT}[r_{xx}(\tau)] = R_{xx}(f) = \frac{1}{\sigma_x^2}X^*(f)X(f) \tag{2.23}$$

This equation states that the auto-correlation function and the (normalized) power spectrum are a Fourier transform pair, also known as the *Wiener–Khinchine Theorem*.

If this relation exists in the continuous system, there should be a similar relationship in the discrete system. Let us calculate the DFT of Eq. (2.16) to see whether the DFT of the auto-correlation function gives the periodogrum.

Thus far, the symbol L was used for the number of averages to calculate the auto-correlation function. In the following discussions, L is changed to N, meaning also the data number of the DFT. In the calculation of the DFT, the parameter n of $x(n)$ must take values 0, 1, 2, …, $(N - 1)$ without skipping a value.

Then, the DFT of the auto-correlation is given by

$$R_{xx}(k) = \mathrm{DFT}[r_{xx}(m)] = \frac{1}{N\sigma_x^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x^*(n)x(n+m)\exp(-j2\pi\frac{mk}{N})$$

By letting $p = n + m \quad (m = p - n)$ and replacing $x(p)$ by $x(p - N)$ (periodicity of $x(p)$), the range of p from $p = n$ to $p = (n + N + 1)$ can be changed to $p = 0$ to $p = N - 1$. Then, the above equation becomes:

$$\begin{aligned} R_{xx}(k) &= \frac{1}{N\sigma_x^2}\sum_{p=0}^{N-1}\sum_{n=0}^{N-1} x^*(n)x(p)\exp\left(-j2\pi\frac{pk}{N}\right)\exp\left(j2\pi\frac{nk}{N}\right) \\ &= \frac{1}{N\sigma_x^2}\sum_{n=0}^{N-1} x^*(n)\exp\left(j2\pi\frac{nk}{N}\right)\sum_{p=0}^{N-1} x(p)\exp(-j2\pi\frac{pk}{N}) \\ &= \frac{1}{N\sigma_x^2}X^*(k)X(k) = \frac{1}{N\sigma_x^2}|X(k)|^2 \end{aligned} \tag{2.24}$$

This is the discrete version of the Wiener–Khinchine Theorem. One reason why the coefficient $1/(N\sigma_x^2)$ is added to the right-hand side is that the auto-correlation function is normalized.

The periodogram of the auto-correlation is calculated using Eq. (2.24)

$$|X(k)|^2 = N\sigma_x^2\mathrm{DFT}[r_{xx}(m)] \tag{2.25}$$

If the auto-correlation function $r_{xx}(m)$ is given for all values of lag m from $-N/2$ to $N/2$, the power spectrum is given by the N-point DFT. Since the auto-correlation is a function of m only, and the power spectrum is independent of the data spacing (P in Eq. (2.8)) used when the auto-correlation function is obtained, Eq. (2.25) is valid independent of the method by which the auto-correlation function is obtained. If the auto-correlation function represents a property of a whole sequence, the power spectrum is also of the same whole sequence.

The IDFT of the (auto) power spectrum must be the auto-correlation function, since the latter is the DFT of the former. Therefore, the auto-correlation can be calculated with less computation steps if the power spectrum is obtained first and then it is inverse-Fourier-transformed by the IFFT. It can be expressed as

$$r_{xx}(m) = \frac{1}{N\sigma_x^2}\mathrm{IFFT}[X^*(k)X(k)] \tag{2.26}$$

However, if $X^*(k)X(k)$ is obtained from one N-point sequence, the auto-correlation function is also given as that of an N-point sequence. In order to obtain $X^*(k)X(k)$ of a whole sequence, N-point power spectra can be calculated by averaging many FFT results with different starting points. But it has been

explained in Chap. 6 in Volume 1 that this does not produce a correct power spectrum.

If the auto-correlation $r_{xx}(m)$ is known for lag $|m| \leq M$ and 0 for $|m| > M$, and if $2M < N$, then the N-point DFT of $r_{xx}(m)$ is exactly equal to the power spectrum of the sequence. In this case, there is no error even if it is assumed that the auto-correlation function is a periodic function with period N.

In the derivation of Eq. (2.24), the relationship that $x(p)$ for $p > N$ is equal to $x(p\text{-}N)$ was used. This assumes that $x(n)$ is periodic with period N. However, this is not correct if an N-point data from a longer sequence $x(n)$ is used. Equation (2.25), which corresponds to the Wiener–Khinchine Theorem (Eq. 2.23) of a continuous system, is valid with the condition that $x(n)$ is a periodic function. In order to calculate the auto-correlation of a long sequence by the DFT, a similar method used for the calculation of convolution must be used to avoid obtaining a circular correlation function. If $x(n)$ is an infinitely long random sequence, since the average of the products of sample values of different data numbers is zero, a circular correlation function will be not obtained. But this is the case for a known spectrum and auto-correlation, and in such a case, there is no need to calculate them.

2.9 Calculation of Auto-Correlation Functions by the FFT

In the previous section, a clue for the calculation of the auto-correlation function by the FFT method was indicated. It is clear that use of the FFT reduces the calculation load, but this process generates a circular DFT. One way to avoid this problem is to append zeros to the data. Another may be to apply appropriate windows. In any case, some degree of error may be unavoidable.

In Fig. 2.13, the handling of data in the auto-correlation calculation by the FFT method is shown. Two sequences $x(n)$ and $y(n)$ shown in the figure are identical. By doing this, it is clear which is the delayed sequence. The sizes of circles are proportional to the data numbers from 1 to 25. The two data sequences with the same sizes of circles are identical data. Figure 2.13 shows the case with a lag of four, and the diagonal lines connecting $x(n)$ and $y(n)$ become vertical if the lag is equal to zero.

In order to calculate the correlation function employing the definition, two sets of data and one with a lag 4 must be selected from a long sequence as shown in Fig. 2.13a. Then they are multiplied and added. Pairs of data used for the multiplication are $x(n)$ and $y(n + 4)$, that are connected by thin lines in the figure.

When the FFT method is used, N-point data within the two vertical dotted lines shown in Fig. 2.13b are used to calculate a periodogram. As explained before, the N-point FFT gives the Fourier coefficients of a sequence with period N. Therefore, the sequences used here are periodic as shown in (b) (in this example, data from 9 to 16 periodically appear). Then, data used by the N-point FFT method are

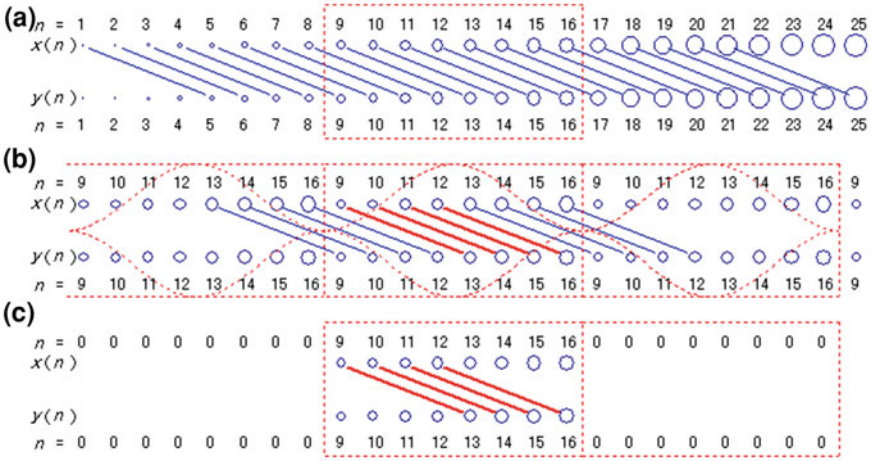


Fig. 2.13 Handling of data for the calculation of auto-correlation functions. However, the method shown by (**c**) still does not give the same result obtained by (**a**). This is because only four correct pairs indicated by *thick lines* in (**c**) are used in the calculation, compared to eight pairs in (**a**). The ratio of these correct pairs to the original pairs decreases linearly as lag m increases. This is the origin of the Bartlet window introduced in Chap. 7 in Volume 1

different from those in (a) except for the pairs connected by the thick diagonal lines. The pairs connected with thin diagonal lines do not exist in the original pairs shown in (a). This causes the difference between auto-correlation functions obtained by the two methods.

In the example with $m = 4$ and $N = 8$ as shown in (b), only four are original (correct) pairs and other four are wrong pairs, indicating that non-negligible errors will be induced. This type of error cannot be ignored except for cases with $m<<N$.

It is possible to avoid the wrong pairs by adding another 8-point zero data to the data from 9 to 16 and using the 2 N-point FFT as shown in (c). Because of the circular correlation property, the data from 1 to 8 are also replaced by zeros. The "wrong" four pairs become 0 and they are excluded from the calculation by the FFT method.

So far, the rectangular window (without weighting) has been used. It was found in Chaps. 6 and 7 that the use of windows such as the Hanning window helps improve the accuracy of spectrum estimation. Our interest is in whether the Hanning window shown in Fig. 2.13b also works to improve the estimation accuracy of the correlation function. As Fig. 2.13b shows, the Hanning window is effective in reducing the values of undesirable pairs shown by thin oblique lines.

Now, four possible methods of calculating the power spectrum are available for the N-point sequence within the vertical dotted lines in Fig. 2.13b. The following four methods are compared.

(0) The power spectrum is obtained by the N-point FFT of the directly calculated correlation function.
(1) The power spectrum is obtained by the N-point FFT of the N-point sequence $x(n)$ using the rectangular window and then the auto-correlation function is calculated by the IFFT.
(2) The power spectrum is obtained by the N-point FFT of the N-point sequence $x(n)$ using the Hanning window and then the auto-correlation function is calculated by the IFFT.
(3) The power spectrum is obtained by the 2 N-point FFT of the N-point sequence $x(n)$ plus N-point zero data and then the auto-correlation function is calculated by the IFFT.

The source signal used here is a convolution of a random sequence and an exponentially decaying impulse response. The reason why the exponentially decaying response is used is so that we can control the lag-dependent decay rate of the auto-correlation function of the convoluted signal. If the impulse response decays quickly, the lag-dependent decay rate of the auto-correlation also decays quickly.

In Fig. 2.14, the top three charts show the random input $r(n)$, impulse response $h(n)$ and the output signal $x(n)$ (convolution of $r(n)$ and $h(n)$), respectively. The impulse response used here is synthesized by the following three steps, (1) add a 2,048 long random sequence and several sine waves, (2) reduce the amplitudes of higher frequency components of the added sequence, and (3) multiply the modified sequence by an exponentially decaying function $e^{-n/50}$ with a time constant of 50.[2]

Auto-correlation functions and power spectra calculated by the four methods are shown in the bottom four rows in Fig. 2.14. In the method (0), the auto-correlation is directly calculated following the definition using all data of $x(n)$. Since the impulse response decays with the time constant of 50, the auto-correlation decays quickly as the lag gets away from 0. The power spectrum is calculated by the FFT using 512 samples of the auto-correlation centered at m (lag) = 0.

In each of the methods from (1) to (3), the power spectrum (periodogram) is obtained as an average of 50 FFTs using data from 50 different parts of $x(n)$, and then, the auto-correlation is calculated by IFFT. Sums of squares of the differences between the auto-correlation functions obtained by the method (1)–(3) and (0) over 512 points centered at $m = 0$ are calculated and their ratio to the energy (sum of squares) of the auto-correlation function of method (0) are also shown in the figure. The method (3) using 2 N-point FFT has the smallest residual power

[2] The time constant is T of a decaying function $e^{-n/T}$, which is the time necessary for the amplitude to decay from 1 to $1/e(e : 2.7828)$.

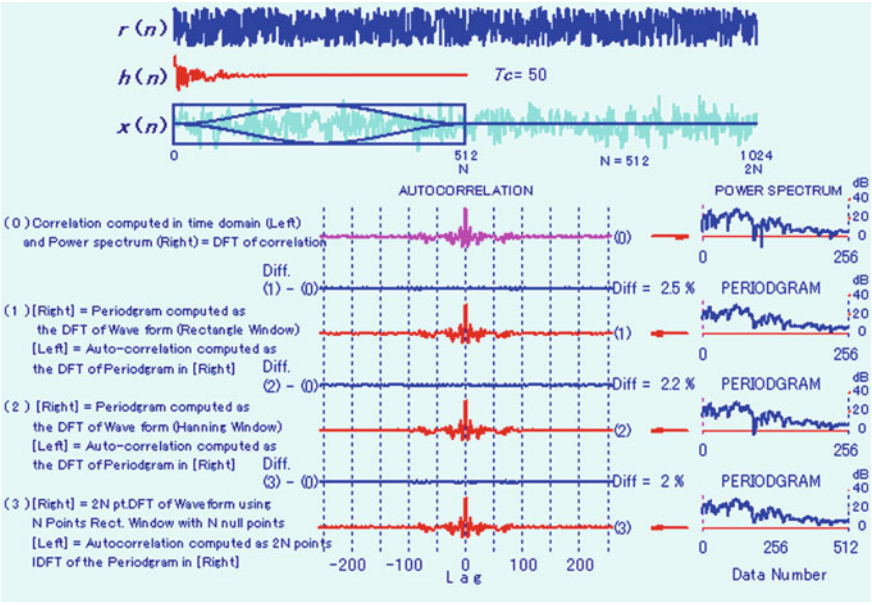


Fig. 2.14 Random input sequence $r(n)$ to a system with an impulse response $h(n)$ and the system output $x(n)$ (*top three charts*) and auto-correlations (*left*) and power spectra (*right*) calculated by the four methods from (0) to (3). Animation available in supplementary files under filename E9-14_AUTOCOR.exe

(energy) ratio, and method (1) has the largest. All three methods have the order of 2 % of the residual power ratio.

The averages of the periodic power spectra obtained by methods (1)–(3) seem to represent the power spectrum obtained by method (0) very well. This is because the spectrum of input sequence $x(n)$ is smooth and the auto-correlation decays quickly as the lag gets away from 0. In method (1), which uses the rectangular window, $x(n + m)$ is replaced by $x(n + m-N)$ when $(n + m)$ exceeds N. In the present case, the correlation between $x(n + m+N)$ and $x(n + m)$ is small and the averages of $x(n + m)x(n + m-N)$ are also small. As a result, the circular correlation problem does not affect the calculated results much when the auto-correlation is calculated by the FFT.

Let us check a case with large auto-correlations for relatively large m. The impulse response $h(n)$ is made of three decaying sine waves with a time constant of 500 sample time intervals. The input is the same random sequence $r(n)$ as in Fig. 2.14. The output signal $x(n)$, the convolution of $r(n)$ and $h(n)$, is used as the signal for the analysis. Those are shown in the top three charts of Fig. 2.15.

The auto-correlation function calculated by method (0) takes on much larger values for large lags than the previous case. The power spectrum obtained by employing the FFT of the 512 point auto-correlation data is shown on the right

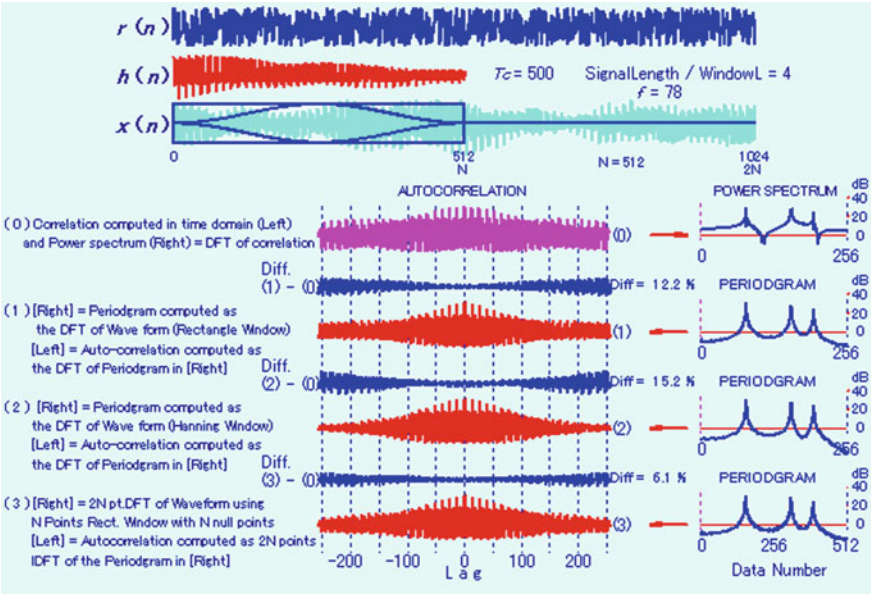


Fig. 2.15 Random input sequence $r(n)$ to a system with an impulse response $h(n)$ and the system output $x(n)$ (*top three charts*) and auto-correlation functions (*left*) and power spectra (*right*) calculated by the four methods from (0) to (3) (four *bottom charts*). Animation available in supplementary files under filename E9-15_AUTOCOR.exe

side. It has three peaks at the frequencies of the sine waves. The power spectrum has large values at frequencies above the highest peak frequency. This is because the auto-correlation function has large values at $m = \pm 256$. The FFT with the rectangular window of this auto-correlation function produces the spreading in the power spectrum.

In methods (1)–(3), the periodic spectra are calculated first and then the auto-correlation functions are obtained by IFFT. The residual power ratio of method (3) is the smallest, but still is over 6 %.

The power spectra of method (0) and (3) look very different, but the residual power ratio is relatively small. This is because the power spectrum is shown in decibels. Even if two values in the low levels have large difference in decibels, the residual power ratio is small if two values around the peaks are small.

2.10 Discussions of Stability and Estimation Precision

First, we will investigate the dependence of the residual power ratio, which was defined in the previous section, on the number of averages.

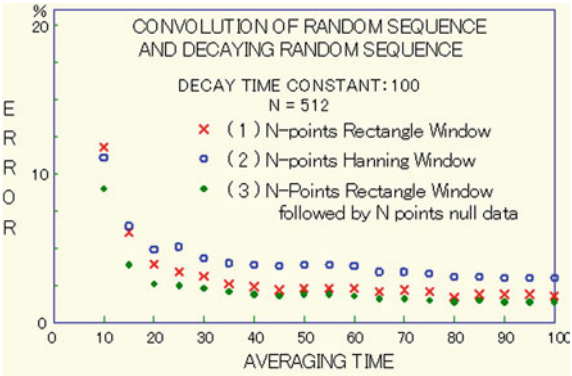


Fig. 2.16 Dependence of residual power ratios of the auto-correlation functions obtained by the three FFT methods on the number of average. The reference power is that of the auto-correlation obtained by the direct method in the time domain

Figure 2.16 shows the results when the number of averages is varied from 5 to 100. As the number of averages increases, the residual power decreases and the rate of change also decreases. The 2 N-point FFT method with the additional N-point zero data gives a smaller residual power ratio than the N-point FFT method with the rectangular and Hanning windows. The 2 N-point FFT method gives residual power ratios lower than 3 % if the number of averages is more than 20.

Shown above is the case with $N = 512$ and the time constant is equal to 100. Let us see how the results change if the time constant is set at different values. The random sequence $x(n)$ is the same as in Fig. 2.15, which is the convolution of the impulse response $h(n)$ comprising three decaying sine waves and the random sequence $r(n)$.

Ratios of the sine wave frequencies are 1:1.9:2.7, and the number of cycles of each wave in the window takes on a non-integer value. The same time constant for these three waves is varied from 10 to 100 in 10 steps. Since the length of $h(n)$ is known to have a very small effect on the results, it is set equal to 5,120. One random sequence generated by a computer is used throughout this example since it also has a very small effect on the result.

Residual power ratios of auto-correlations calculated by the three FFT methods to auto-correlations calculated by the direct time domain method are shown in Fig. 2.17. The results for (1) N-point FFT with the rectangular window, (2) N-point FFT with the Hanning window, and (3) 2 N-point FFT with additional N-point zero data are shown by $\times$, ○, and ●, respectively. Residual power ratios for one random sequence $x(n)$ were calculated as functions of the decay constant. The eight curves for each case shown in the figure were obtained for eight different random sequences.

The results show the following. All three methods have small residual powers if the time constant is below 50 (less than 1/10 of the window length). This is a proof that the effect of the circular correlation (replacement of $x(n + m)$ by $x(n + m-N)$ when $n + m$ exceeds N) is small because of the small correlations for m (lag) $> N$ if the time constant is much shorter than the window length.

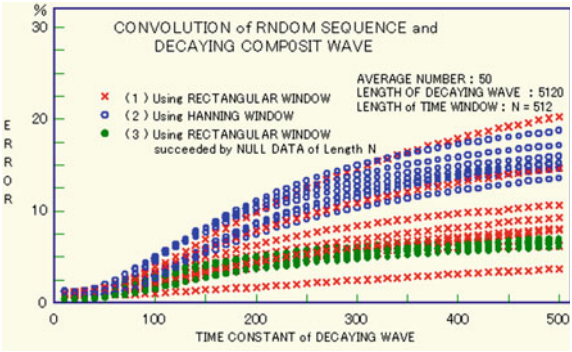


Fig. 2.17 Dependence of residual power ratios of auto-correlations obtained by the three FFT methods on the time constant

For large time constants, the correlations for large lags become large and the effect of the circular correlation becomes non-negligible. In this case, the differences of the residual powers among the three methods become visible.

In the first method, using the N-point FFT with the rectangular window, the residual power ratio changes drastically depending on the frequencies of the sine waves as shown by $\times$ in Fig. 2.17. If the frequencies of the sine waves are kept constant, the residual power ratio changes smoothly as the time constant increases. However, in this method, the starting and ending values of $x(n)$ of each window have large effects on this curve and, therefore, the curves for different sequences take quite different paths.

The second method, using the Hanning window to reduce the starting and ending values near the edges of the window, may be a good choice for the reduction of the unstable results of the first method. The curves shown by ○ are the results of the second method. The variation of the results is reduced but the magnitudes of the residual powers are not small compared to other two methods. This is because the auto-correlations for larger lags sharply reduce due to the window, compared to those of the direct time domain method.

The above mentioned variations are removed in the third method since $x(n + m)$ becomes always zero when $(n + m)$ exceeds N. The value of auto-correlation reduces linearly from $m = 0$ to $m = \pm N$. Accordingly, the residual power ratio gets as large as approximately 1/12 even for large time constants (see Appendix 2D). Curves indicated by ● in the figure show this. Small variations still exist since the random sequences are different for different calculations.

The conclusion from the above discussion is that the IDFT of the averaged power spectrum obtained by the 2 N-point FFT method with the additional N-point zero data gives a good and stable estimation of the true auto-correlation function. The averaged power spectrum obtained by this method is a good estimation of the true power spectrum but still some amount of bias error cannot be avoided.

The time domain calculation of the auto-correlation function follows the formula Eq. (2.12). On the other hand, the FFT methods use the finite sequence lengths, N or 2 N. Data used in the direct method and the FFT methods are

different, which is the reason the auto-correlation functions obtained by the FFT methods are different from the one obtained by the direct method. This becomes more evident when the ratio of the time constant to the window length gets relatively large.

In order to handle data with the same length, finite sequences with length N with external zero data may be used in the direct method. The auto-correlation functions obtained this way are different from that of the long sequence but they may be more preferable if time dependent properties of data are investigated.

2.11 DFT of Cross-Correlation Functions

Calculations of the cross-correlation functions and the related functions are similar to those of the auto-correlation functions. So, let us start from the calculation of the DFT of the cross-correlation function defined as Eq. (2.20). Since the cross-correlation function is obtained from N-point sequences by the same procedure as the auto-correlation function, the DFT of the cross-correlation function is calculated similarly as

$$R_{xy}(k) = \mathrm{DFT}[r_{xy}(m)] = \frac{1}{N\sigma_x\sigma_y}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x^*(n)y(n+m)\exp(-j2\pi\frac{mk}{N})$$

The above equation is rewritten as (see Appendix 2E)

$$R_{xy}(k) = \frac{1}{N\sigma_x\sigma_y}X^*(k)Y(k) \tag{2.27}$$

or

$$X^*(k)Y(k) = N\sigma_x\sigma_y R_{xy}(k) \tag{2.28}$$

Equation (2.27) means that the DFT of the cross-correlation function between $x(n)$ and $y(n)$ is equal to $X^*(k)Y(k)$, the complex conjugate of $X(k)$ (DFT of $x(n)$) multiplied by $Y(k)$ (DFT of $y(n)$) and divided by $N\sigma_x\sigma_y$. The product of $X*(k)$ and $Y(k)$ is the cross-power spectrum, which will be discussed in Chap. 3.

The IDFT of Eq. (2.27) gives Eq. (2.29) which enables the calculation of the cross-correlation function from the cross-power spectrum.

$$r_{xy}(m) = \frac{1}{N\sigma_x\sigma_y}\mathrm{IDFT}[X^*(k)Y(k)] \tag{2.29}$$

The cross-power spectrum in this equation is the one calculated by the DFT of the cross-correlation function, therefore, Eq. (2.29) should not be interpreted as an equation that gives the cross-correlation function from the cross-power spectrum.

It must be considered carefully whether the cross-power spectrum obtained by the DFTs of sequences can be applied to the right-hand side of Eq. (2.29) to give the cross-correlation function.

The cross-power spectrum $X^*(k)Y(k)$ calculated by N-point DFTs is complex and its real and imaginary parts take both positive and negative values. The values given by averaging many cross-power spectra of two uncorrelated sequences will approach zero as the number of averages increases. If we consider only this aspect, it seems that the cross-power spectrum obtained by employing many averages can be used for the right-hand side of Eq. (2.29). This is different from the case of the auto-power spectrum.

The circular correlation problem of the DFT also exists in the cross-power spectrum. If there is a way of avoiding or reducing this, it may be possible to replace $X^*(k)Y(k)$ by using many averages of the cross-power spectrum obtained by selecting continuous N-point sequences from $x(n)$ and y(n).

$$r_{xy}(m) = \frac{1}{N\sigma_x\sigma_y}\,\mathrm{IDFT}\left[\overline{X^*(k)Y(k)}\right] \tag{2.30}$$

The bar above $X^*(k)Y(k)$ indicates that many averages have been taken. By employing this equation it becomes possible to calculate the cross-correlation function by using the N-point FFTs and IFFTs of the two sequences with much fewer computations.

If $x(t)$ and $y(t)$ are continuous time functions, the relation shown below exists between the cross-correlation and the cross-power spectrum (see Appendix 2F).

$$R_{xy}(\tau) = \frac{1}{\sigma_x\sigma_y}X^*(f)Y(f) \tag{2.31}$$

If a random sequence $x(n)$ goes through a transfer system and results in $y(n)$ as its output, the cross-correlation between $x(n)$ and $y(n)$ is the impulse response $h(n)$ of the system. A simulation example is shown in Fig. 2.4. A random sequence $x(n)$ is input to the same transfer system with the impulse response $h(n)$, and the cross-correlation function between the input $x(n)$ and the output $y(n)$ is calculated. The results are shown in Fig. 2.18. The top three charts are for $x(n)$, $h(n)$ and $y(n)$, respectively. The left side of Fig. 2.18(0) shows the cross-correlation function directly calculated in the time domain after 1,000 averages. In this simulation, the spacing between adjacent sets of sample sequences are varied between 7 and 19. The DFT of the cross-correlation function is shown by the right side of Fig. 2.18(0). This corresponds to Eq. (2.27).

The cross-power spectrum is a function made from a real even function and an imaginary odd function. They are shown separately in the right side of the figure only in the positive frequency region. Residual power ratio is also shown at the right corner of each correlation function (impulse responses). The residual power ratio is the sum of the squares of differences between the sample values of the estimated and the true impulse responses to the sum of the squares of the sample

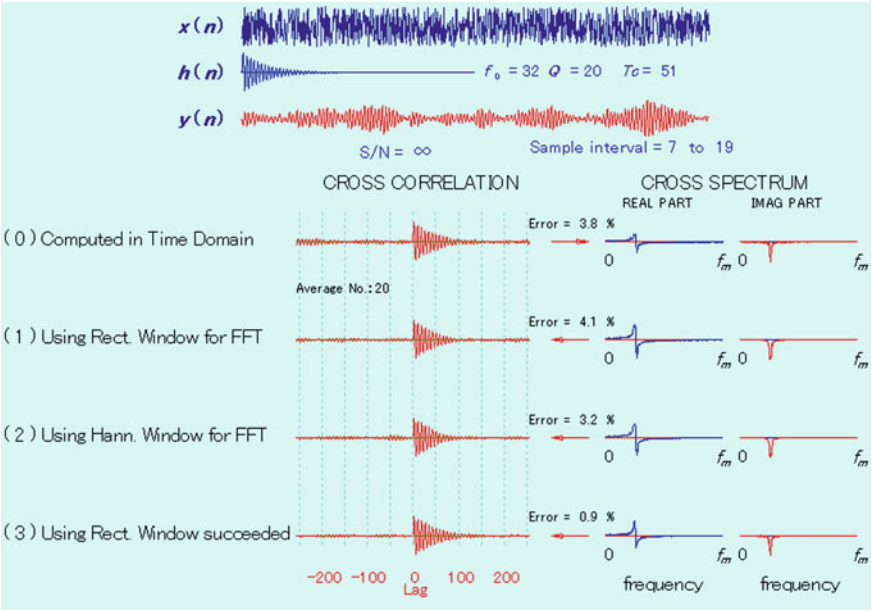


Fig. 2.18 Random input sequence $x(n)$ and its output $y(n)$ through a system with an impulse response $h(n)$ (*top three charts*) and the cross-correlation functions and corresponding cross-power spectra obtained by four methods (*bottom four columns*). Animation available in supplementary files under filename E9-18_CrosCorFT.exe

values of the true impulse response in the positive time region. Fig. 2.18(0) shows the case when the cross-correlation function is calculated in the time domain. The residual power ratio is 3.9 %.

The next three results are obtained using Eq. (2.30) in a similar manner to those discussed in Sects. 2.9 and 2.10. The residual power ratios of Fig. 2.18(1–3) are 4.1, 3.2 and 0.9 %, respectively. The last method (3), that avoids the circular correlation problem, gives much smaller errors than the other methods.

The cross-correlation between the input and output becomes the impulse response of the system only when the auto-correlation function of the input sequence is equal to one at m (lag) $= 0$ and 0 elsewhere. If not, the impulse response cannot be estimated from the cross-correlation function.

Figure 2.19 shows one example of this case. The input sequence of this example is a 4-point moving average of a random sequence. Only the input is changed from Fig. 2.18. It can be seen that the residual power ratios are much larger than the previous case. This difference is caused by the averaging of the input sequence only over the 4-point width.

It was shown in Fig. 2.11 that the auto-correlation of a moving-averaged random sequence becomes an isosceles triangle with a base that has twice the width of

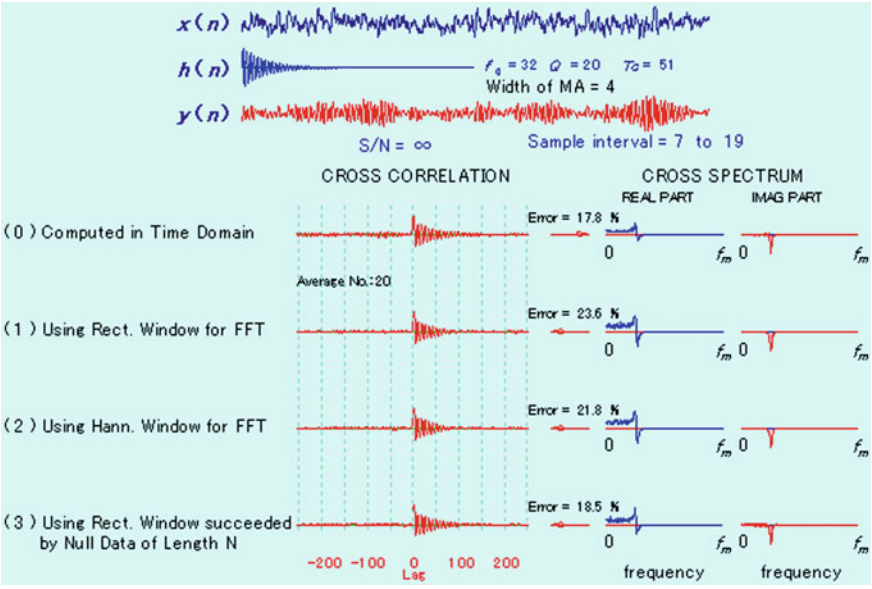


Fig. 2.19 Random input sequence $x(n)$ with 4-point moving average and its output $y(n)$ through a system with an impulse response $h(n)$ (*top* three charts) and the cross-correlation functions and corresponding cross-power spectra obtained by four methods (*bottom* four columns). Animation available in supplementary files under filename E9-19_CrosCorFT.exe

the averaging. In order for the cross-correlation function to be the impulse response, the width of the isosceles triangle must be infinitely narrow. In other words, the input must be perfectly uncorrelated (auto-correlation must be zero except at $m = 0$). This condition cannot always be expected. If the estimation of the impulse response is an objective, we should consider other methods. That will be discussed in Chap. 3.

In the programs associated with the figures, the student can choose various conditions and numbers. By trying various cases, the student will learn more that is not discussed in this chapter.

2.12 Exercise

1. What is the standard deviation of a sine function with amplitude 1?
2. Why is Eq. (2.4) equal to two when x(n) and y(n) are completely uncorrelated?
3. Why is Eq. (2.4) equal to four when x(n) and y(n) have the same magnitudes but their signs are opposite?

4. Prove for the case of a continuous system that the correlation function between the input and output functions becomes the impulse response of the system if the input sequence is a random sequence.
5. One tried to determine the system impulse response from the cross-correlation between the input and the system output by using a random input sequence. However, the input contained a DC component. Is it still possible to obtain the impulse response?
6. Discuss the relationship between the time constant and the period of half decay.
7. What is the auto-correlation function of a rectangular pulse with amplitude one from $t = 0$ to T and 0 elsewhere?
8. What is the auto-correlation function of one period of a sine wave from $t = 0$ and T and 0 elsewhere?
9. What is the cross-correlation function of the two functions given in problem 7 and problem 8?
10. What is the result if the frequency of the sine wave increases by 50 %?
11. What is the cross-correlation function between a function with one period of a sine wave within $t = 0$ and T and 0 elsewhere and a function with one period of a cosine wave within $t = 0$ and T and 0 elsewhere?
12. What is the result if the numbers of waves in the period from 0 to T are multiplied by 1.5?
13. What is the result if the number of sine wave in the period from 0 to T is multiplied by 1.5 and the number of the cosine wave in the same period is multiplied by 2.3?
14. When the cross-correlation is calculated using Eq. (2.8), there is a risky case if P is constant. What is it?

http://www.springer.com/978-1-4939-1126-4

Digital Fourier Analysis: Advanced Techniques
Kido, K.
2015, XIII, 178 p. 100 illus. in color., Softcover
ISBN: 978-1-4939-1126-4