

Contents

Part I Basics

1	Parallel Programming Paradigms	3
1.1	Computational Models	3
1.1.1	Performance Metrics	3
1.1.2	Single Instruction Multiple Data Architectures and Pipelining	5
1.1.3	Multiple Instruction Multiple Data Architectures	9
1.1.4	Hierarchical Architectures	10
1.2	Principles of Parallel Programming	11
1.2.1	From Amdahl's Law to Scalability	12
	References	14
2	Fundamental Kernels	17
2.1	Vector Operations	17
2.2	Higher Level BLAS	19
2.2.1	Dense Matrix Multiplication	20
2.2.2	Lowering Complexity via the Strassen Algorithm	22
2.2.3	Accelerating the Multiplication of Complex Matrices	24
2.3	General Organization for Dense Matrix Factorizations	25
2.3.1	Fan-Out and Fan-In Versions	25
2.3.2	Parallelism in the Fan-Out Version	26
2.3.3	Data Allocation for Distributed Memory	28
2.3.4	Block Versions and Numerical Libraries	29
2.4	Sparse Matrix Computations	30
2.4.1	Sparse Matrix Storage and Matrix-Vector Multiplication Schemes	31
2.4.2	Matrix Reordering Schemes	36
	References	43

Part II Dense and Special Matrix Computations

3	Recurrences and Triangular Systems	49
3.1	Definitions and Examples	49
3.2	Linear Recurrences	51
3.2.1	Dense Triangular Systems	52
3.2.2	Banded Triangular Systems	57
3.2.3	Stability of Triangular System Solvers	59
3.2.4	Toeplitz Triangular Systems	61
3.3	Implementations for a Given Number of Processors	66
3.4	Nonlinear Recurrences	72
	References	78
4	General Linear Systems	79
4.1	Gaussian Elimination	80
4.2	Pairwise Pivoting	82
4.3	Block LU Factorization	84
4.3.1	Approximate Block Factorization	85
4.4	Remarks	87
	References	88
5	Banded Linear Systems	91
5.1	LU-based Schemes with Partial Pivoting	91
5.2	The Spike Family of Algorithms	94
5.2.1	The Spike Algorithm	95
5.2.2	Spike: A Polyalgorithm	99
5.2.3	The Non-diagonally Dominant Case	100
5.2.4	The Diagonally Dominant Case	104
5.3	The Spike-Balance Scheme	105
5.4	A Tearing-Based Banded Solver	115
5.4.1	Introduction	115
5.4.2	Partitioning	115
5.4.3	The Balance System	118
5.4.4	The Hybrid Solver of the Balance System	124
5.5	Tridiagonal Systems	126
5.5.1	Solving by Marching	128
5.5.2	Cyclic Reduction and Parallel Cyclic Reduction	130
5.5.3	LDU Factorization by Recurrence Linearization	139
5.5.4	Recursive Doubling	143
5.5.5	Solving by Givens Rotations	144
5.5.6	Partitioning and Hybrids	149
5.5.7	Using Determinants and Other Special Forms	155
	References	159

6	Special Linear Systems	165
6.1	Vandermonde Solvers	166
6.1.1	Vandermonde Matrix Inversion	170
6.1.2	Solving Vandermonde Systems and Parallel Prefix	172
6.1.3	A Brief Excursion into Parallel Prefix	174
6.2	Banded Toeplitz Linear Systems Solvers	176
6.2.1	Introduction	176
6.2.2	Computational Schemes	182
6.3	Symmetric and Antisymmetric Decomposition (SAS)	192
6.3.1	Reflexive Matrices as Preconditioners	194
6.3.2	Eigenvalue Problems	196
6.4	Rapid Elliptic Solvers	197
6.4.1	Preliminaries	198
6.4.2	Mathematical and Algorithmic Infrastructure	199
6.4.3	Matrix Decomposition	201
6.4.4	Complete Fourier Transform	203
6.4.5	Block Cyclic Reduction	205
6.4.6	Fourier Analysis-Cyclic Reduction	210
6.4.7	Sparse Selection and Marching	211
6.4.8	Poisson Inverse in Partial Fraction Representation	214
6.4.9	Notes	218
	References	220
7	Orthogonal Factorization and Linear Least Squares Problems	227
7.1	Definitions	227
7.2	QR Factorization via Givens Rotations	228
7.3	QR Factorization via Householder Reductions	232
7.4	Gram-Schmidt Orthogonalization	233
7.5	Normal Equations Versus Orthogonal Reductions	235
7.6	Hybrid Algorithms When $m \gg n$	236
7.7	Orthogonal Factorization of Block Angular Matrices	237
7.8	Rank-Deficient Linear Least Squares Problems	242
	References	246
8	The Symmetric Eigenvalue and Singular-Value Problems	249
8.1	The Jacobi Algorithms	251
8.1.1	The Two-Sided Jacobi Scheme for the Symmetric Standard Eigenvalue Problem	251
8.1.2	The One-Sided Jacobi Scheme for the Singular Value Problem	254
8.1.3	The Householder-Jacobi Scheme	259
8.1.4	Block Jacobi Algorithms	261
8.1.5	Efficiency of Parallel Jacobi Methods	262

8.2	Tridiagonalization-Based Schemes.	263
8.2.1	Tridiagonalization of a Symmetric Matrix	264
8.2.2	The QR Algorithm: A Divide-and-Conquer Approach.	265
8.2.3	Sturm Sequences: A Multisectioning Approach.	267
8.3	Bidiagonalization via Householder Reduction.	272
	References.	273

Part III Sparse Matrix Computations

9	Iterative Schemes for Large Linear Systems	277
9.1	An Example.	278
9.2	Classical Splitting Methods	280
9.2.1	Point Jacobi	281
9.2.2	Point Gauss-Seidel	282
9.2.3	Line Jacobi	283
9.2.4	Line Gauss-Seidel	286
9.2.5	The Symmetric Positive Definite Case.	287
9.3	Polynomial Methods	294
9.3.1	Chebyshev Acceleration.	294
9.3.2	Krylov Methods	299
	References.	309
10	Preconditioners	311
10.1	A Tearing-Based Solver for Generalized Banded Preconditioners	312
10.2	Row Projection Methods for Large Nonsymmetric Linear Systems.	312
10.2.1	The Kaczmarz Scheme	313
10.2.2	The Cimmino Scheme	319
10.2.3	Connection Between RP Systems and the Normal Equations	319
10.2.4	CG Acceleration	320
10.2.5	The 2-Partitions Case	321
10.2.6	Row Partitioning Goals	324
10.2.7	Row Projection Methods and Banded Systems	325
10.3	Multiplicative Schwarz Preconditioner with GMRES.	326
10.3.1	Algebraic Domain Decomposition of a Sparse Matrix	327
10.3.2	Block Multiplicative Schwarz	329
10.3.3	Block Multiplicative Schwarz as a Preconditioner for Krylov Methods.	336
	References.	340

11 Large Symmetric Eigenvalue Problems	343
11.1 Computing Dominant Eigenpairs and Spectral Transformations	343
11.1.1 Spectral Transformations	345
11.1.2 Use of Sturm Sequences	349
11.2 The Lanczos Method	350
11.2.1 The Lanczos Tridiagonalization	350
11.2.2 The Lanczos Eigensolver	352
11.3 A Block Lanczos Approach for Solving Symmetric Perturbed Standard Eigenvalue Problems	356
11.3.1 Starting Vectors for $A(S_i)x = \lambda X$	356
11.3.2 Starting Vectors for $A(S_i)^{-1}x = \mu x$	358
11.3.3 Extension to the Perturbed Symmetric Generalized Eigenvalue Problems	359
11.3.4 Remarks	360
11.4 The Davidson Methods	363
11.4.1 General Framework	363
11.4.2 Convergence	364
11.4.3 Types of Correction Steps	366
11.5 The Trace Minimization Method for the Symmetric Generalized Eigenvalue Problem	368
11.5.1 Derivation of the Trace Minimization Algorithm	370
11.5.2 Practical Considerations	373
11.5.3 Acceleration Techniques	378
11.5.4 A Davidson-Type Extension	381
11.5.5 Implementations of TRACEMIN	384
11.6 The Sparse Singular-Value Problem	386
11.6.1 Basics	386
11.6.2 Subspace Iteration for Computing the Largest Singular Triplets	390
11.6.3 The Lanczos Method for Computing a Few of the Largest Singular Triplets	392
11.6.4 The Trace Minimization Method for Computing the Smallest Singular Triplets	395
11.6.5 Davidson Methods for the Computation of the Smallest Singular Values	398
11.6.6 Refinement of Left Singular Vectors	399
References	402

Part IV Matrix Functions and Characteristics

12 Matrix Functions and the Determinant	409
12.1 Matrix Functions.	410
12.1.1 Methods Based on the Product Form of the Denominator	412
12.1.2 Methods Based on Partial Fractions.	414
12.1.3 Partial Fractions in Finite Precision	418
12.1.4 Iterative Methods and the Matrix Exponential.	424
12.2 Determinants	428
12.2.1 Determinant of a Block-Tridiagonal Matrix	429
12.2.2 Counting Eigenvalues with Determinants	431
References.	434
13 Computing the Matrix Pseudospectrum	439
13.1 Grid Based Methods	440
13.1.1 Limitations of the Basic Approach	440
13.1.2 Dense Matrix Reduction	442
13.1.3 Thinning the Grid: The Modified GRID Method.	443
13.2 Dimensionality Reduction on the Domain: Methods Based on Path Following	446
13.2.1 Path Following by Tangents	446
13.2.2 Path Following by Triangles.	450
13.2.3 Descending the Pseudospectrum	454
13.3 Dimensionality Reduction on the Matrix: Methods Based on Projection	458
13.3.1 An EIGTOOL Approach for Large Matrices	459
13.3.2 Transfer Function Approach	460
13.4 Notes	463
References.	464
Index	467

Parallelism in Matrix Computations

Gallopoulos, E.; Philippe, B.; Sameh, A.H.

2016, XXX, 473 p. 58 illus., Hardcover

ISBN: 978-94-017-7187-0