Contents

Volume I

Foreword xv
About This Manual xix
Customer Support xxv

Chapter 1 Introduction to CMSSL for CM Fortran 1

1.1 About CMSSL 1
1.2 Contents of CMSSL for CM Fortran 2
1.2.1 Library Routines 2
1.2.2 Nodal Library Restrictions 27
1.2.3 Safety Mechanism 27
1.3 Notes on Terminology and Conventions 28
1.3.1 Data Types 28
1.3.2 Array Axis Descriptions 28
1.3.3 Coordinates Are One-Based 28
1.3.4 Processing Elements and Subgrids 29
1.4 Data Types Supported 30
1.4.1 Coordinates Must Be Single-Precision Integers 30
1.4.2 Support for Double-Precision Integers 30
1.5 Support for Multiple Instances 33
1.5.1 Defining Multiple Independent Data Sets 36
1.5.2 Notation Used for CM Arrays and Embedded Matrices 37
1.5.3 Rules for Data Axes and Instance Axes 38
1.5.4 Specifying Single-Instance vs. Multiple-Instance Operations . 38
1.6 Numerical Stability for the Linear Algebra Routines 42
1.7 Numerical Complexity 43
1.8 CM Fortran Performance Enhancements with CMSSL 50

Copyright © 1994 Thinking Machines Corporation

Chapter 2 Using the CM Fortran CMSSL Interface 53

2.1 Creating a CM Fortran CMSSL Program 53
2.1.1 Including the CMSSL Header File 54
2.1.2 Calling CMSSL Routines 54
2.1.3 Compiling and Linking 54
2.1.4 Executing CMSSL Programs 56

2.2 A Note about Aligning Arrays 56

2.3 Using the CMSSL Safety Mechanism 57
2.3.1 Safety Mechanism Features 57
2.3.2 Levels of Error Checking 58
2.3.3 Setting the CMSSL Safety Environment Variable 59
2.3.4 Using CMSSL Safety from within a Program 60

2.4 On-Line Sample Code and Man Pages 60

Chapter 3 Dense Matrix Operations 63

3.1 Inner Product 64
Man Page for Inner Product 66

3.2 2-Norm 73
Man Page for 2-Norm 74

3.3 Outer Product 77
Man Page for Outer Product 78

3.4 Matrix Vector Multiplication 82
Man Page for Matrix Vector Multiplication 83

3.5 Vector Matrix Multiplication 87
Man Page for Vector Matrix Multiplication 88

3.6 Infinity Norm 92
Man Page for Infinity Norm 93

3.7 Matrix Multiplication 96
Man Page for Matrix Multiplication 97

3.8 Matrix Multiplication with External Storage 103
Man Page for Matrix Multiplication with External Storage 104

3.9 References 107

Version 3.2, April 1994
Copyright © 1994 Thinking Machines Corporation

Chapter 4 Sparse Matrix Operations 109

4.1 Introduction 109
4.1.1 Arbitrary Sparse Matrix Operations 110
4.1.2 Grid Sparse Matrix Operations 113

4.2 Arbitrary Elementwise Sparse Matrix Operations 114
4.2.1 The Arbitrary Elementwise Sparse Matrix Routines 114
4.2.2 Storage of Sparse Matrices 115
4.2.3 Communication Algorithms 116
4.2.4 Permutation of Source and Destination Array Element Locations 117

Man Page for Arbitrary Elementwise Sparse Matrix Operations 121

4.3 Arbitrary Block Sparse Matrix Operations 133
4.3.1 The Arbitrary Block Sparse Matrix Routines 133
4.3.2 Block Representation, Gathering, and Scattering 133
4.3.3 Communication Algorithms 135
4.3.4 Random Permutation of Source and Destination Array Element Locations 136
4.3.5 Examples 137

Man Page for Arbitrary Block Sparse Matrix Operations 147

4.4 Grid Sparse Matrix Operations 159
4.4.1 The Grid Sparse Matrix Routines 159
4.4.2 Grid Sparse Matrix Representation 159
4.4.3 Matrix Representation of the Grid Sparse Matrix Operations 165

Man Page for Grid Sparse Matrix Operations 171

4.5 References 178

Chapter 5 Linear Solvers for Dense Systems 179

5.1 Introduction 179
5.1.1 Embedding Coefficient Matrices within Larger Matrices 181
5.1.2 Choosing an Algorithm 181

5.2 Gaussian Elimination ("*LU*" Routines) 183
5.2.1 The *LU* Routines 183
5.2.2 Blocking and Load Balancing 186
5.2.3 The *LU* Factors Defined 186
5.2.4 Results of the Solver Routines 190
5.2.5 Numerical Stability 191
5.2.6 Saving and Restoring the *LU* State 191

Man Page for Gaussian Elimination 193

Copyright © 1994 Thinking Machines Corporation

5.3 Routines for Solving Linear Systems Using Householder Transformations ("*QR*" Routines) 215
5.3.1 The *QR* Routines 216
5.3.2 Householder Algorithm 219
5.3.3 Blocking and Load Balancing 222
5.3.4 The *QR* Factors Defined 223
5.3.5 Choosing a Backsolve Strategy 229
5.3.6 Results of the Solver Routines 232
5.3.7 Numerical Stability 232
5.3.8 The Pivoting Option: Working with Ill-Conditioned Systems . 233
5.3.9 Scaling 236
5.3.10 Saving and Restoring the *QR* State 237
Man Page for Solving Linear Systems Using Householder Transformations 239
5.4 Matrix Inversion and the Gauss-Jordan System Solver 264
5.4.1 Matrix Inversion 264
5.4.2 The Gauss-Jordan Solver 266
5.4.3 Stability and Performance 266
Man Page for Matrix Inversion 267
Man Page for Gauss-Jordan System Solver 270
5.5 Gaussian Elimination with External Storage 273
Man Page for Gaussian Elimination with External Storage 274
5.6 *QR* Factorization and Least Squares Solution with External Storage . . . 284
Man Page for *QR* Factorization and Least Squares Solution with External Storage 285
5.7 References 290

Chapter 6 Linear Solvers for Banded Systems 293

6.1 Banded System Factorization and Solver Routines (Unified) 295
6.1.1 The Routines and Their Functions 295
6.1.2 Algorithms Used 296
6.1.3 How to Set Up Your Data 304
6.1.4 Need for Interface Blocks 310
Man Page for Banded System Factorization and Solver Routines (Unified) 313
6.2 Banded System Factorization and Solver Routines 322
Man Page for Banded System Factorization and Solver Routines 323
6.3 References 333

Version 3.2, April 1994
Copyright © 1994 Thinking Machines Corporation

Chapter 7 Iterative Solvers 335

7.1 Krylov-Based Iterative Solvers 335
7.1.1 CMSSL Iterative Solver Routines 335
7.1.2 Algorithms 336
7.1.3 Acknowledgments 337
7.1.4 Example 337

Man Page for Iterative Solvers 340

7.2 References 350

Chapter 8 Eigensystem Analysis 353

8.1 Introduction 355

8.2 Reduction to Tridiagonal Form and Corresponding Basis Transformation 359
8.2.1 Blocking and Load Balancing 359
8.2.2 Numerical Stability 359

Man Page for Reduction to Tridiagonal Form and Corresponding Basis Transformation 360

8.3 Eigenvalues of Real Symmetric Tridiagonal Matrices 366
8.3.1 Parallel Bisection Algorithm 366
8.3.2 Accuracy 367
8.3.3 Restriction 367

Man Page for Eigenvalues of Real Symmetric Tridiagonal Matrices 369

8.4 Eigenvectors of Real Symmetric Tridiagonal Matrices 371
8.4.1 Inverse Iteration Algorithm 371
8.4.2 Accuracy 371
8.4.3 Applicability 372
8.4.4 Restriction 373
8.4.5 Performance 373

Man Page for Eigenvectors of Real Symmetric Tridiagonal Matrices . . . 374

8.5 Eigensystem Analysis of Dense Hermitian Matrices 377
8.5.1 Accuracy 377

Man Page for Eigensystem Analysis of Dense Hermitian Matrices 378

8.6 Generalized Eigensystem Analysis of Dense Hermitian Matrices 382
8.6.1 Accuracy 383

Man Page for Generalized Eigensystem Analysis of Dense Hermitian Matrices 384

Copyright © 1994 Thinking Machines Corporation

8.7 Eigensystem Analysis of Real Symmetric Matrices Using Jacobi Rotations 388
8.7.1 Accuracy 388
Man Page for Eigensystem Analysis of Real Symmetric Matrices Using Jacobi Rotations 390
8.8 Selected Eigenvalue and Eigenvector Analysis Using a *k*-Step Lanczos Method 394
8.8.1 The *k*-Step Lanczos Algorithm 394
8.8.2 Input Arguments and Data Structures 395
8.8.3 Multiple Eigenvalues 396
8.8.4 Convergence Properties and Spectral Transformations 396
8.8.5 Reverse Communication Interface 397
8.8.6 Data Layout Guidelines 401
8.8.7 On-Line Example 401
8.8.8 Acknowledgments 402
Man Page for Selected Eigenvalue and Eigenvector Analysis Using a *k*-Step Lanczos Method 403
8.9 Selected Eigenvalue and Eigenvector Analysis Using a k-Step Arnoldi Method 413
8.9.1 The *k*-Step Arnoldi Algorithm 413
8.9.2 Input Arguments and Data Structures 414
8.9.3 Reverse Communication Interface 415
8.9.4 Data Layout Guidelines 418
8.9.5 On-Line Example 418
8.9.6 Acknowledgments 419
Man Page for Selected Eigenvalue and Eigenvector Analysis Using a *k*-Step Arnoldi Method 420
8.10 References 431

Chapter 9 Singular Value Decomposition 433

9.1 Bidiagonalization and Corresponding Basis Transformation 435
9.1.1 Load Balancing 435
9.1.2 Numerical Stability 435
Man Page for Bidiagonalization and Corresponding Basis Transformation 436
9.2 Singular Values of Bidiagonal Matrices 447
9.2.1 Implementation 447
Man Page for Singular Values of Bidiagonal Matrices 448
9.3 Singular Vectors of Bidiagonal Matrices 451

Copyright © 1994 Thinking Machines Corporation

9.3.1 Implementation 451
Man Page for Singular Vectors of Bidiagonal Matrices 452
9.4 Singular Value Decomposition of Dense Real Matrices 456
Man Page for Singular Value Decomposition of Dense Real Matrices 457
9.5 References 463

Index **465**

Volume II

About This Manual xv
Customer Support xxi

Chapter 10 Fast Fourier Transforms **475**
10.1 Introduction 476
10.1.1 The CMSSL FFT Library Calls 477
10.2 Complex-to-Complex FFT 479
10.2.1 Butterfly Computations, Twiddle Factors, and the CCFFT Setup Phase 479
10.2.2 Bit Ordering and Bit Reversal 479
10.2.3 Multidimensional and Multiple-Instance FFTs 481
10.2.4 Current Restrictions 482
10.2.5 Implementation and Performance 483
10.2.6 Efficient Computation of Convolutions and Correlations 487
10.2.7 For Users Familiar with the CM-200 FFT 488
Man Page for Complex-to-Complex Fast Fourier Transform 490
10.3 Real-to-Complex and Complex-to-Real FFTs 499
10.3.1 Relationship of CRFFT, RCFFT, and CCFFT 500
10.3.2 Data Orderings 500
10.3.3 Multidimensional and Multiple-Instance Real-to-Complex and Complex-to-Real FFTs 503
10.3.4 Implementation 505
10.3.5 Steps in Performing the Real-to-Complex and Complex-to-Real FFTs 509

Copyright © 1994 Thinking Machines Corporation

Man Page for Real-to-Complex and Complex-to-Real Fast Fourier Transform 511

10.4 Array Conversion Utilities for the Real-to-Complex and Complex-to-Real FFTs 521

Man Page for Array Conversion Utilities for the Real-to-Complex and Complex-to-Real FFTs 522

10.5 References 526

Chapter 11 Ordinary Differential Equations **529**

11.1 Explicit Integration of Ordinary Differential Equations Using a Runge-Kutta Method 529

11.1.1 Examples 530

Man Page for Explicit Integration of Ordinary Differential Equations Using a Runge-Kutta Method 532

11.2 References 539

Chapter 12 Linear Programming **541**

12.1 Dense Simplex Routine 541

12.1.1 Geometrical Description of the Algorithm 541

12.1.2 Vertices and Bases 542

12.1.3 Input Array Format 542

12.1.4 Reinversion 543

12.1.5 Degeneracy 544

12.1.6 Implementation 544

12.1.7 Example 544

Man Page for Dense Simplex 546

12.2 References 553

Chapter 13 Random Number Generators **555**

13.1 Introduction 555

13.1.1 The Fast RNG and the VP RNG Compared 556

13.1.2 The RNG Routines 557

13.1.3 Implementation 558

13.2 State Tables 559

13.2.1 Fast RNG State Tables 559

13.2.2 VP RNG State Tables 560

13.2.3 State Table Parameters 561

13.2.4 Need for Deallocation 562

Copyright © 1994 Thinking Machines Corporation

13.2.5 Parameters Saved During Checkpointing 562
13.3 Safety Checkpointing 564
13.4 Alternate-Stream Checkpointing 565
13.5 Fast RNG 568
Man Page for Fast RNG 569
13.6 VP RNG 577
Man Page for VP RNG 578
13.7 References 587

Chapter 14 Statistical Analysis **589**
14.1 How to Histogram 590
Man Page for Histogram 592
Man Page for Range Histogram 594

Chapter 15 Communication Primitives **597**
15.1 Polyshift 599
15.1.1 The Polyshift Routines 599
15.1.2 Optimization Recommendations 601
Man Page for Polyshift 602
15.2 All-to-All Rotation 609
15.2.1 The All-to-All Rotation Routines 611
Man Page for All-to-All Rotation 613
15.3 All-to-All Broadcast 620
Man Page for All-to-All Broadcast 621
15.4 All-to-All Reduction 625
Man Page for All-to-All Reduction 626
15.5 Matrix Transpose 631
15.5.1 Example 631
15.5.2 Special Performance Note for Three-Dimensional and Higher-Rank Arrays 631
Man Page for Matrix Transpose 633
15.6 Sparse Gather Utility 636
15.6.1 The Gather Utility Routines 636
15.6.2 Definition of the Gather Operation 636
15.6.3 Gather Operation Examples 637

Copyright © 1994 Thinking Machines Corporation

Man Page for Sparse Gather Utility 640
15.7 Sparse Scatter Utility 644
15.7.1 The Scatter Utility Routines 644
15.7.2 Definition of the Scatter Operation 644
15.7.3 Scatter Operation Example 645
Man Page for Sparse Scatter Utility 647
15.8 Sparse Vector Gather Utility 651
15.8.1 Definition of the Vector Gather Operation 651
15.8.2 Examples 651
Man Page for Sparse Vector Gather Utility 654
15.9 Sparse Vector Scatter Utility 658
15.9.1 Definition of the Vector Scatter Operation 658
15.9.2 Example 658
Man Page for Sparse Vector Scatter Utility 660
15.10 Block Gather and Scatter Utilities 664
Man Page for Block Gather and Scatter Utilities 667
15.11 Partitioning of an Unstructured Mesh and Reordering of Pointers 671
15.11.1 Definitions 672
15.11.2 Finite Element Numbering Scheme 674
15.11.3 Partitioning Rules 675
15.11.4 The Partitioning Permutation 679
15.11.5 Reordering a Pointers Array 680
15.11.6 Renumbering a Pointers Array 681
Man Page for Partitioning of an Unstructured Mesh and Reordering of Pointers 683
15.12 Partitioned Gather Utility 693
Man Page for Partitioned Gather Utility 694
15.13 Partitioned Scatter Utility 700
Man Page for Partitioned Scatter Utility 701
15.14 Communication Compiler 707
15.14.1 Communication Compiler Routines 707
15.14.2 How to Use the Communication Compiler 709
Man Page for Communication Compiler 711
15.15 Vector Move (Extract and Deposit) 724
Man Page for Vector Move (Extract and Deposit) 725
15.16 Computation of Block Cyclic Permutations 729
15.16.1 Blocking, Load Balancing, and Block Cyclic Ordering 730

Version 3.2, April 1994
Copyright © 1994 Thinking Machines Corporation

15.16.2 Obtaining *L*, *U*, and *R* Factors in Elementwise Consecutive Order 732

Man Page for Computation of Block Cyclic Permutations 734

15.17 Permutation Along an Axis 737

Man Page for Permutation Along an Axis 738

15.18 Combination of Permutations 741

Man Page for Combination of Permutations 742

15.19 Zeroing Routine 744

Man Page for Zeroing Routine 745

15.20 Send-to-NEWS and NEWS-to-Send Reordering 746

Man Page for Send-to-NEWS and NEWS-to-Send Reordering 747

15.21 References 749

Appendix A CMSSL and LAPACK: Corresponding Routines **751**

Index **761**

Copyright © 1994 Thinking Machines Corporation