About This Manual

Objectives of This Manual

This manual describes the CM Fortran programming interface to Version 3.2 of the Connection Machine Scientific Software Library (CMSSL) on the Connection Machine supercomputer, model CM-5.

Intended Audience

Anyone writing CM Fortran programs that use CMSSL software on the CM-5 should read this document.

Revision Information

This manual supersedes the manual entitled *CMSSL for CM Fortran: CM-5 Edition*, Version 3.1, dated June 1993.

Organization of This Manual

This manual is divided into two volumes with fifteen chapters and an appendix:

Volume I

Chapter 1 Introduction to CMSSL for CM Fortran
Describes the contents of CMSSL. Discusses the data types supported and explains how to perform CMSSL operations on multiple independent data sets concurrently.

Chapter 2 Using the CMSSL CM Fortran Interface
Explains how to include CMSSL routine definitions in CM Fortran code, and how to compile, link, and execute CM Fortran programs that call CMSSL routines.

Copyright © 1994 Thinking Machines Corporation

Chapter 3 Dense Matrix Operations
Describes the inner product, 2-norm, outer product, matrix vector multiplication, vector matrix multiplication, matrix multiplication, and infinity norm routines. Also describes the routine that performs matrix multiplication routine with external storage.

Chapter 4 Sparse Matrix Operations
Describes the routines that perform arbitrary elementwise sparse matrix operations, arbitrary block sparse matrix operations, and grid sparse matrix operations.

Chapter 5 Linear Solvers for Dense Systems
Describes the in-core linear solvers: Gaussian elimination (*LU* decomposition) routines, routines that solve linear systems using Householder transformations (the *QR* routines), matrix inversion, the Gauss-Jordan system solver. Also describes the external (out-of-core) Gaussian elimination and *QR* factorization routines.

Chapter 6 Linear Solvers for Banded Systems
Describes the banded system factorization and solver routines, which solve tridiagonal, block tridiagonal, pentadiagonal, and block pentadiagonal systems.

Chapter 7 Iterative Solvers
Describes routines that solve linear systems using Krylov space iterative methods.

Chapter 8 Eigensystem Analysis
Describes routines that perform eigensystem analysis of real symmetric tridiagonal systems, dense Hermitian systems, dense real symmetric systems, dense real systems, and sparse systems. Included are routines that use the Jacobi method, a *k*-step Lanczos method, and a *k*-step Arnoldi method. Also included are routines that reduce Hermitian matrices to real symmetric tridiagonal form (and perform the corresponding basis transformation).

Copyright © 1994 Thinking Machines Corporation

Chapter 9 Singular Value Decomposition
Describes routines that perform singular value decomposition of real bidiagonal systems and dense real systems. Included are routines that bidiagonalize dense real matrices (and perform the corresponding basis transformation).

Volume II

Chapter 10 Fast Fourier Transforms
Describes the simple and detailed complex-to-complex FFT routines; the real-to-complex and complex-to-real FFT routines; and array conversion utilities for the real-to-complex and complex-to-real FFTs.

Chapter 11 Ordinary Differential Equations
Describes routines that integrate ordinary differential equations (ODEs) explicitly using a fifth-order Runge-Kutta-Fehlberg formula.

Chapter 12 Linear Programming
Describes a routine that solves multi-dimensional minimization problems using the simplex linear programming method.

Chapter 13 Random Number Generators
Describes the Fast and VP random number generators.

Chapter 14 Statistical Analysis
Describes the histogram and range histogram routines.

Copyright © 1994 Thinking Machines Corporation

Chapter 15 Communication Primitives
Describes the polyshift operation; the all-to-all rotation, broadcast, and reduction routines; a matrix transpose routine; the sparse gather and scatter, sparse vector gather and scatter, and block gather and scatter utilities; partitioning of an unstructured mesh and reordering of pointers; the partitioned gather and scatter utilities; the communication compiler; the vector move (extract and deposit) routines; routines that compute block cyclic permutations, permute an array along an axis, combine permutations, and zero an array; and send-to-NEWS and NEWS-to-send reordering routines.

Appendix A CMSSL and LAPACK: Corresponding Routines
Lists the LAPACK routines that have CMSSL equivalents. LAPACK is a library of Fortran 77 subroutines for solving numerical linear algebra problems.

Acknowledgments

The Arnoldi routines (Chapter 8) are the result of collaborative work between Thinking Machines Corporation and the Center for Research on Parallel Computation at Rice University. The bidiagonalization and singular value decomposition routines (Chapter 9) are the result of collaborative development with the Danish Computing Center for Research and Education (UNI−C).

Related Documents

For more detailed information about CM Fortran, consult the latest versions of the books listed below.

- *Getting Started in CM Fortran*

  Offers a brief introduction to using the CM Fortran language.

- *CM Fortran Programming Guide*

  Offers a more detailed, task-oriented introduction to all the major features of the CM Fortran language.

Version 3.2, April 1994
Copyright © 1994 Thinking Machines Corporation

- *CM Fortran User's Guide*

  Includes complete descriptions of how to compile, link, and execute CM Fortran code, as well as how to use the CM Fortran Utility Library.

- *CM Fortran Utility Library Reference Manual*

  Provides reference and usage information about the procedures in the CM Fortran Utility Library.

- *CM Fortran Reference Manual*

  The definitive description of the CM Fortran language.

Notation Conventions

The table below displays the notation conventions used in this manual.

Convention	Meaning
`bold typewriter`	UNIX and CMOST commands, command options, and filenames, when they appear embedded in text.
`typewriter`	Code examples and code fragments.
`% ` **`bold typewriter`** `typewriter`	In interactive examples, user input is shown in bold typewriter and system output is shown in regular typewriter font.
boldface sans serif	CM Fortran language elements, such as function and subroutine names and constants, when they appear embedded in text or in syntax lines.
bold italics	CM arrays, when they appear embedded in text or syntax lines.
italics	Argument names (except CM arrays), when they appear embedded in text or syntax lines.

Copyright © 1994 Thinking Machines Corporation

Standard Abbreviations for Matrix Operations and Matrix Types

The following standard abbreviations are used in CMSSL CM Fortran interfaces to identify matrix types. Further abbreviations will be introduced as more matrix types are supported.

CMSSL Matrix Type Abbreviations

dense general	**gen**
dense symmetric	**sym**
arbitrary elementwise sparse	**sparse**
arbitrary block sparse	**block_sparse**
grid sparse	**grid_sparse**
bidiagonal	**gen_bidiag**
tridiagonal	**gen_tridiag**
pentadiagonal	**gen_pentadiag**
block tridiagonal	**block_tridiag**
block pentadiagonal	**block_pentadiag**

The following standard abbreviations are used in the CMSSL CM Fortran interfaces to identify matrix operations:

CMSSL Matrix Operation Abbreviations

factorization	**factor**
inversion	**invert**
multiplication	**mult**
solver	**solve**
polyshift	**pshift**

Copyright © 1994 Thinking Machines Corporation