Release 3 News

1 Introduction

Release 3 represents a significant expansion of the NAG Parallel Library. It contains a total of 183 documented routines, of which 95 are new at this release.

The new facilities include a new Chapter, C06, containing routines for discrete Fourier transforms in one, two and three dimensions; considerable additional functionality in the area of the solution of sparse linear equations, for example, complex equivalents of the real solvers and further preconditioners; ScaLAPACK routines for the solution of banded and tridiagonal linear systems, and for the solution of symmetric and Hermitian eigenproblems; black box solvers for the solution of banded and tridiagonal linear systems; random number generators for uniform, Normal and exponential distributions; utility routines to aid in operations such the generation and distribution of dense and sparse matrices, and vectors distributed conformally to a sparse matrix; and utility routines to gather and output dense matrices and vectors distributed conformally to a sparse matrix.

2 New Routines

The 95 new user callable routines included in the NAG Parallel Library at Release 3 are as follows:

C06FUFP	Direct or inverse two-dimensional Fourier transform of a complex sequence
C06GXFP	Factorizes a positive integer n as $n = n1 \times n2$. This routine may be used in conjunction with C06MCFP
C06MCFP	Direct or inverse one-dimensional discrete Fourier transform of a complex seqence
C06MXFP	Direct or inverse three-dimensional discrete Fourier transform of a complex sequence
F01CPFP	Element-wise maximum or minimum in absolute value of integer matrices
F01WAFP	Gather real matrix, regarded as submatrix of matrix distributed in cyclic two-dimensional block format, used with routines from Chapters F07 and F08
F01WBFP	Gather real matrix distributed in cyclic two-dimensional block format, used with routines from Chapter F04
F01WGFP	Gather complex matrix distributed in cyclic two-dimensional block format, used with routines from Chapters F07 and F08
F01WHFP	Gather complex matrix distributed in cyclic two-dimensional block format, used with routines from Chapter F04
F01WNFP	Scatter real matrix from the root processor to the Library Grid using cyclic two-dimensional block format, used with routines from Chapters F07 and F08
F01WPFP	Scatter real matrix from the root processor to the Library Grid using cyclic two-dimensional block format, used with routines from Chapter F04
F01WUFP	Scatter complex matrix from the root processor to the Library Grid using cyclic two-dimensional block format, used with routines from Chapters F07 and F08
F01WVFP	Scatter complex matrix from the root processor to the Library Grid using cyclic two-dimensional block format, used with routines from Chapter F04
F01XAFP	Scatter real sparse matrix, stored in coordinate storage format, using cyclic row block distribution
F01XEFP	Scatter real vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01XFFP	Gather real vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01XGFP	Scatter integer vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01XHFP	Gather integer vector, distributed conformally to sparse matrix, used with routines from Chapter F11
F01XPFP	Scatter complex sparse matrix, stored in coordinate storage format, using cyclic row block distribution, used with routines from Chapter F11

F01XTFP	Scatter complex vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01XUFP	Gather complex vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01YBFP	In-place generation of real sparse matrix using cyclic row block distribution (suitable for repeated sparsity pattern), used with routines from Chapter F11
F01YPFP	In-place generation of complex sparse matrix according to cyclic row block distribution, used with routines from Chapter F11
F01YQFP	In-place generation of complex sparse matrix according to cyclic row block distribution (suitable for repeated sparsity pattern)
F01YTFP	In-place generation of complex dense vector distributed conformally to sparse matrix, used with routines from Chapter F11
F01YWFP	In-place generation of complex Hermitian banded matrix in column block fashion, used with routines from Chapter F07
F01YXFP	In-place generation of real symmetric banded matrix in column block fashion, used with routines from Chapter F07
F01YYFP	In-place generation of real matrix in row block fashion on a one-dimensional grid of processors, used with routines from Chapter F07
F01YZFP	In-place generation of complex matrix in row block fashion on a one-dimensional grid of processors, used with routines from Chapter F07
F01ZHFP	Generates an l by m by n three-dimensional array $\mathrm{A}(i,j,k)$ on a grid of processors in i-block form
F01ZMFP	In-place generation of real matrix in row block fashion, used with routines from Chapters C06 and F04
F01ZNFP	In-place generation of complex matrix in row block fashion, used with routines from Chapter F04
F01ZYFP	In-place generation of complex vector in column block fashion, used with routines from Chapter F07
F01ZZFP	In-place generation of real vector in column block fashion, used with routines from Chapter F07
F04HBFP	Solution of real symmetric banded linear system (Black Box)
F04HZFP	Solution of complex Hermitian banded linear system (Black Box)
F04JBFP	Solution of real symmetric tridiagonal linear system (Black Box)
F04JZFP	Solution of complex Hermitian tridiagonal linear system (Black Box)
F07HDFP	(PDPBTRF) Cholesky factorization of real symmetric banded matrix with no pivoting
F07HEFP	(PDPBTRS) Solution of real symmetric banded linear system, matrix already factorized by F07HDFP (PDPBTRF)
F07HRFP	(PZPBTRF) Cholesky factorization of complex Hermitian banded matrix with no-pivoting
F07HSFP	(PZPBTRS) Solution of complex Hermitian banded linear system, matrix already factorized by F07HRFP (PZPBTRF)
F07JDFP	(PDPTTRF) Cholesky factorization of real symmetric tridiagonal matrix with no-pivoting
F07JEFP	(PDPTTRS) Solution of real symmetric tridiagonal linear system, matrix already factorized by F07JDFP (PDPTTRF)
F07JRFP	(PZPTTRF) Factorization of complex Hermitian tridiagonal matrix with no-pivoting
F07JSFP	(PZPTTRS) Solution of real symmetric tridiagonal linear system, matrix already factorized by F07JRFP (PZPTTRF)
F08FGFP	(PDORMTR) Apply orthogonal transformation determined by F08FEFP (PDSYTRD)
F08FSFP	(PZHETRD) Unitary reduction of complex Hermitian matrix to real symmetric tridiagonal form
F08FUFP	(PZUNMTR) Apply unitary transformation matrix determined by F08FSFP (PZHETRD)
F08JKFP	(PDSTEIN) Selected eigenvectors of real symmetric tridiagonal matrix by inverse iteration, storing eigenvectors in real array
F08JXFP	(PZSTEIN) Selected eigenvectors of real symmetric tridiagonal matrix by inverse iteration, storing eigenvectors in complex array

F11BRFP	Complex sparse non-Hermitian linear system, reverse-communication, set-up for F11BSFP
F11BSFP	Complex sparse non-Hermitian linear system, reverse-communication, solver using preconditioned GMRES, CGS or Bi-CGSTAB
F11BTFP	Complex sparse non-Hermitian linear system, reverse-communication, diagnostic for F11BSFP
F11DDFP	Apply iterations of SOR method to real sparse linear system, used mostly as SOR preconditioner for real sparse matrix
F11DEFP	Solution of real sparse nonsymmetric linear system using Jacobi, SOR or no preconditioned RGMRES, CGS or Bi-CGSTAB (Black Box)
F11DKFP	Apply iterations of relaxed Jacobi iterative method to a real sparse linear system, used mostly as Jacobi preconditioner for real sparse matrix
F11DRFP	Apply iterations of SOR method to the complex sparse linear system, used mostly as SOR preconditioner for complex sparse matrix
F11DSFP	Solution of complex sparse non-Hermitian linear system using Jacobi, SOR or no preconditioned RGMRES, CGS or Bi-CGSTAB (Black Box)
F11DTFP	Computes incomplete *LU* factorization of local diagonal blocks of complex sparse matrix
F11DUFP	Complex sparse non-Hermitian linear system, reverse-communication, block-Jacobi preconditioner for complex sparse matrix
F11DVFP	Solution of complex sparse non-Hermitian linear system using block Jacobi preconditioned RGMRES, CGS or Bi-CGSTAB (Black Box)
F11DXFP	Apply iterations of relaxed Jacobi iterative method to complex sparse linear system, used mostly as Jacobi preconditioner for complex sparse matrix
F11JEFP	Solution of real sparse symmetric linear system using Jacobi, SSOR or no preconditioned CG or SYMMLQ (Black Box)
F11JHFP	Solution of sparse symmetric linear system using block Jacobi preconditioned CG or SYMMLQ (Black Box)
F11XPFP	Matrix-vector multiplication for complex sparse matrix
F11YAFP	Permutation of non-zero entries of real sparse matrix with repeated sparsity pattern
F11YBFP	Permutation of real vector from distribution based order to local indexing based order
F11YCFP	Permutation of real vector from local indexing based order to distribution based order
F11YNFP	Permutation of non-zero entries of complex sparse matrix with repeated sparsity pattern
F11YPFP	Permutation of complex vector from distribution based order to local indexing based order
F11YQFP	Permutation of complex vector from local indexing based order to distribution based order
F11ZBFP	General set-up routine for real sparse matrix distributed in cyclic row block form (suitable for repeated sparsity pattern)
F11ZGFP	Generates multi-colour ordering for real sparse matrix with symmetric sparsity pattern, distributed in row block form
F11ZPFP	General set-up routine for complex sparse matrix, distributed in cyclic row block form (suitable for repeated sparsity pattern)
F11ZUFP	Generates multi-colour ordering for complex sparse matrix with symmetric sparsity pattern, distributed in row block form.
F11ZZFP	Release of internally allocated memory
G05ACFP	Function returning pseudo-random real number from the interval $[a,b)$, uniform distribution
G05ADFP	Function returning pseudo-random real number from the interval $[a,b)$, Normal distribution
G05AEFP	Function returning pseudo-random real number from the interval $[a,b)$, exponential distribution
G05AZFP	Function returning pseudo-random integer from the interval $[ia,ib)$, uniform distribution

G05BAFP	Pseudo-random real numbers from the interval (0,0), uniform distribution
G05BBFP	Selects random number generator and initialises seeds to give repeatable sequence
G05BCFP	Pseudo-random real numbers from the interval (a,b), uniform distribution
G05BDFP	Pseudo-random real numbers from the interval (a,b), Normal distribution
G05BEFP	Pseudo-random real numbers from the interval (a,b), exponential distribution
G05BZFP	Pseudo-random integers from the interval (ia,ib), uniform distribution
X04BMFP	Outputs set of general integer matrices distributed on a two-dimensional logical processor grid
X04BXFP	Outputs real matrix stored in row block fashion
X04BZFP	Outputs complex matrix stored in row block fashion (complex version of X04BXFP)
X04YPFP	Outputs complex vector, distributed conformally to sparse matrix to a sequential file
Z01CFFP	Computes number of rows of a row block distributed matrix owned by a processor
Z01ZAFP	Returns information on coordinates in Library Grid set up by Z01AAFP
Z01ZBFP	Creates an MPI communicator from a Library context
Z02EAFP	Specification of error checking level, can reduce the amount of checking carried out in subsequent calls to other Library routines

3 Renamed Routines

Three routines from Release 2 of the NAG Parallel Library have been renamed at Release 3 for compatibility with the NAG Fortran Library:

Release 2	**Release 3**
F11DAFP	F11DFFP
F11DBFP	F11DGFP
F11DCFP	F11DHFP

The user is advised to read the new document.

4 Withdrawn Routines

Release 3 of the NAG Parallel Library only supports the use of MPI as the message passing system, and in consequence three utility routines concerned with support for PVM have been withdrawn.

Withdrawn Routine	**Recommended Replacement**
Z01ADFP	Z01AEFP
Z01BDFP	Z01BGFP
Z01BFFP	None

5 Routines Scheduled for Withdrawal

The routines listed below are scheduled for withdrawal from the NAG Parallel Library, because improved routines have been included in the Library. In keeping with the current release of ScaLAPACK, all the ScaLAPACK routines (Chapters F07 and F08) that include the arguments LWORK or LIWORK now have the facility to query how large workspace should be by setting LWORK or LIWORK to -1 on input, consequently the corresponding Z01 routines are now redundant. The following routines will be withdrawn at Release 4.

Routines Scheduled for Withdrawal	**Recommended Replacement**
F11XAFP	F11XBFP
F11ZAFP	F11ZBFP
Z01CBFP	Set LWORK = -1 in call to F08AEFP or F08AFFP
Z01CCFP	Set LWORK = -1 in call to F08AGFP
Z01CEFP	Set LWORK = -1 in call to F08FEFP