Matrix-Vector and Matrix-Matrix Operations

Prototype name: `FLA_XXYY`

Matrix types (XX)

Ge	– General
Sy	– Symmetric
Tr	– Triangular

Operations (YY)

mv	– matrix-vector multiply
r	– rank-1 update
r2	– rank-2 update
sv	– solve
mm	– matrix-matrix multiply
rk	– rank-k update
r2k	– rank-2k update
sm	– solve with multiple right-hand sides

Options

trans_	- FLA_NO_TRANSPOSE FLA_TRANSPOSE FLA_CONJ_TRANSPOSE
uplo	- FLA_UPPER_TRIANGULAR FLA_LOWER_TRIANGULAR
diag	- FLA_NONUNIT_DIAG FLA_UNIT_DIAG FLA_ZERO_DIAG
side	- FLA_LEFT FLA_RIGHT (A or op(A) on the left or on the right of B)

Constants (can be used for parameters alpha or beta)

-1	- FLA_MINUS_ONE
0	- FLA_ZERO
1	- FLA_ONE

Output

All routines return integer constant FLA_SUCCESS or FLA_FAILURE.

Funding

The FLAME project has been sponsored in part by numerous NSF grants as well as grants and gifts from industry. For additional funding information visit http://www.cs.utexas.edu/users/flame/.

References

C. Lawson, R. Hanson, D. Kincaid, and F. Krogh, "Basic Linear Algebra Subprograms for Fortran Usage," *ACM Trans. on Math. Soft.* 5 (1979) 308-325

J.J. Dongarra, J. DuCroz, S. Hammerling, and R. Hanson, "An Extended Set of Fortran Basic Linear Algebra Subprograms," *ACM Trans. on Math. Soft.* 14, 1 (1988) 1-32

J.J. Dongarra, J. DuCroz, I. Duff, and S. Hammarling, "A Set of Level 3 Basic Linear Algebra Subprograms," *ACM Trans. on Math. Soft.* 16, 1 (1990) 1-17

Paolo Bientinesi, Enrique Quintana-Orti, and Robert van de Geijn, "Representing Linear Algebra Algorithms in Code: The FLAME APIs." *ACM Trans. on Math. Soft.* 31, 1 (2005) 27-59

Field G. Van Zee, *libflame: The Complete Reference.* www.lulu.com, *2009.* www.cs.utexas.edu/users/field/docs/libflame.pdf.

This brochure was patterned after similar brochures produced as part of the LAPACK project.

More Information

http://www.cs.utexas.edu/users/flame/

Example

(Unblocked Cholesky factorization)

```
int Chol_unb_var3( FLA_Obj A )
{
  FLA_Obj ATL,   ATR,      A00,  a01,     A02,
          ABL,   ABR,      a10t, alpha11, a12t,
                           A20,  a21,     A22;

  FLA_Part_2x2( A,    &ATL, &ATR,
                      &ABL, &ABR,   0, 0, FLA_TL );

  while ( FLA_Obj_length(ATL)<FLA_Obj_length(A) ){

    FLA_Repart_2x2_to_3x3(
       ATL, ATR,      &A00,  &a01,     &A02,
                      &a10t, &alpha11, &a12t,
       ABL, ABR,      &A20,  &a21,     &A22,
                           1, 1, FLA_BR );
    /*------------------------------------------*/

    if ( FLA_Sqrt( alpha11 ) != FLA_SUCCESS )
       return FLA_FAILURE;
    FLA_Inv_scal( alpha11, a21 );
    FLA_Syr( FLA_LOWER_TRIANGULAR,
             FLA_MINUS_ONE, a21, A22 );

    /*------------------------------------------*/
    FLA_Cont_with_3x3_to_2x2(
       &ATL, &ATR,   A00,  a01,     A02,
                     a10t, alpha11, a12t,
       &ABL, &ABR,   A20,  a21,     A22, FLA_TL );
  }
  return FLA_SUCCESS;
}
```

Generate code skeleton with Spark:
http://www.cs.utexas.edu/users/flame/Spark/

FLAME

C Language Interface

Basic

Linear

Algebra

Subprograms

A Quick Reference Guide

MISCELLANEOUS OPERATIONS

`FLA_Copy ( A, B )`	$B \leftarrow A$
`FLA_Copyt ( trans, A, B )`	$B \leftarrow \mathrm{trans}(A)$
`FLA_Swap ( A, B )`	$B \leftrightarrow A$
`FLA_Swapt ( trans, A, B )`	$B \leftrightarrow \mathrm{trans}(A)$
`FLA_Axpy ( alpha, A, B )`	$B \leftarrow \alpha A + B$
`FLA_Axpyt ( trans, alpha, A, B )`	$B \leftarrow \alpha\, \mathrm{trans}(A) + B$
`FLA_Scal ( alpha, A )`	$A \leftarrow \alpha A$
`FLA_Inv_scal( alpha, A )`	$A \leftarrow A / \alpha$

VECTOR-VECTOR OPERATIONS

`FLA_Dot ( x, y, delta )`	$\delta \leftarrow x^T y$
`FLA_Dots ( alpha, x, y, beta, delta )`	$\delta \leftarrow \alpha x^T y + \beta \delta$
`FLA_Nrm2 ( x, delta )`	$\delta \leftarrow \lVert x \rVert_2$
`FLA_Asum ( x, delta )`	$\delta \leftarrow \lVert x \rVert_1$
`FLA_Iamax ( x, i )`	$i \leftarrow 1^{st}\ k$ s.t. $\lvert x_k \rvert = \max_j (\lvert x_j \rvert)$

MATRIX-VECTOR OPERATIONS

`FLA_Gemv ( trans, alpha, A, x, beta, y )`	$y \leftarrow \alpha A x + \beta y,\ y \leftarrow \alpha A^T x + \beta y,\ y \leftarrow \alpha A^H x + \beta y$
`FLA_Symv ( uplo, alpha, A, x, beta, y )`	$y \leftarrow \alpha A x + \beta y$
`FLA_Trmv ( uplo, trans, diag, A, y )`	$y \leftarrow A y,\ y \leftarrow A^T y,\ y \leftarrow A^H y$
`FLA_Trmvsx ( uplo, trans, diag, alpha, A, x, beta, y )`	$y \leftarrow \alpha A x + \beta y,\ y \leftarrow \alpha A^T x + \beta y,\ y \leftarrow \alpha A^H x + \beta y$
`FLA_Trsv ( uplo, trans, diag, A, y )`	$y \leftarrow A^{-1} y,\ y \leftarrow A^{-T} y,\ y \leftarrow A^{-H} y$
`FLA_Trsvsx ( uplo, trans, diag, alpha, A, x, beta, y )`	$y \leftarrow \alpha A^{-1} x + \beta y,\ y \leftarrow \alpha A^{-T} x + \beta y,\ y \leftarrow \alpha A^{-H} x + \beta y$
`FLA_Ger ( alpha, x, y, A )`	$A \leftarrow \alpha x y^T + A$
`FLA_Syr ( uplo, alpha, x, A )`	$A \leftarrow \alpha x x^T + A$
`FLA_Syr2 ( uplo, alpha, x, y, A )`	$A \leftarrow \alpha (x y^T + y x^T) + A$

MATRIX-MATRIX OPERATIONS

`FLA_Gemm ( transA, transB, alpha, A, B, beta, C )`	$C \leftarrow \alpha\, \mathrm{op}(A)\, \mathrm{op}(B) + \beta C,\ \mathrm{op}(X) = X, X^T, X^H$
`FLA_Symm ( side, uplo, alpha, A, B, beta, C )`	$C \leftarrow \alpha A B + \beta C,\ C \leftarrow \alpha B A + \beta C,\ A = A^T$
`FLA_Syrk ( uplo, trans, alpha, A, beta, C )`	$C \leftarrow \alpha A A^T + \beta C,\ C \leftarrow \alpha A^T A + \beta C,\ C = C^T$
`FLA_Syr2k ( uplo, trans, alpha, A, B, beta, C )`	$C \leftarrow \alpha (A B^T + B A^T) + \beta C,\ C \leftarrow \alpha (A^T B + B^T A) + \beta C,\ C = C^T$
`FLA_Trmm ( side, uplo, trans, diag, alpha, A, B )`	$B \leftarrow \alpha\, \mathrm{op}(A)\, B,\ B \leftarrow \alpha B\, \mathrm{op}(A),\ \mathrm{op}(X) = X, X^T, X^H$
`FLA_Trmmsx ( side, uplo, trans, diag, alpha, A, B, beta, C )`	$C \leftarrow \alpha\, \mathrm{op}(A)\, B + \beta C,\ C \leftarrow \alpha B\, \mathrm{op}(A) + \beta C,\ \mathrm{op}(X) = X, X^T, X^H$
`FLA_Trsm ( side, uplo, trans, diag, alpha, A, B )`	$B \leftarrow \alpha\, \mathrm{op}(A^{-1})\, B,\ B \leftarrow \alpha B\, \mathrm{op}(A^{-1}),\ \mathrm{op}(X) = X, X^T, X^H$
`FLA_Trsmsx ( side, uplo, trans, diag, alpha, A, B, beta, C )`	$C \leftarrow \alpha\, \mathrm{op}(A^{-1})\, B + \beta C,\ C \leftarrow \alpha B\, \mathrm{op}(A^{-1}) + \beta C,\ \mathrm{op}(X) = X, X^T, X^H$

For additional routines available consult Field Van Zee. libflame: The Complete Reference. Available from http://www.lulu.com or download from http://www.cs.utexas.edu/users/field/docs/libflame.pdf.