Cholesky factorization/decomposition

Prof. **Paolo Bientinesi**

pauldj@aices.rwth-aachen.de

May 29, 2018

HPAC High Performance and Automatic Computing

RWTH AACHEN UNIVERSITY

High-level description

$$LL^T = A \qquad\qquad L := \Gamma(A)$$

$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array} \right) = ?$$

High-level description

$$LL^T = A \qquad L := \Gamma(A)$$

$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) = \,?$$

$$\left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) \left(\begin{array}{c|c} L_{TL}^T & L_{BL}^T \\ \hline & L_{BR}^T \end{array}\right) = \left(\begin{array}{c|c} A_{TL} & A_{BL}^T \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

High-level description

$$LL^T = A \qquad\qquad L := \Gamma(A)$$

$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) = ?$$

$$\left(\begin{array}{c|c} L_{TL}L_{TL}^T = A_{TL} & \\ \hline L_{BL}L_{TL}^T = A_{BL} & L_{BL}L_{BL}^T + L_{BR}L_{BR}^T = A_{BR} \end{array}\right)$$

High-level description

$$LL^T = A \qquad\qquad L := \Gamma(A)$$

$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) = ?$$

Partitioned Matrix Expression (PME):

$$\left(\begin{array}{c|c} L_{TL} = \Gamma(A_{TL}) & \\ \hline L_{BL} = A_{BL} L_{TL}^{-T} & L_{BR} = \Gamma\big(A_{BR} - L_{BL} L_{BL}^T\big) \end{array}\right)$$

High-level description

$$LL^T = A \qquad\qquad L := \Gamma(A)$$

$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) = ?$$

Operations:

$$\left(\begin{array}{c|c} 1)\ L_{TL} = \text{CHOL} & \\ \hline 2)\ L_{BL} = \text{TRSM} & 3)\ L_{BR} = \text{CHOL(SYRK)} \end{array}\right)$$

High-level description

$$LL^T = A \qquad\qquad L := \Gamma(A)$$

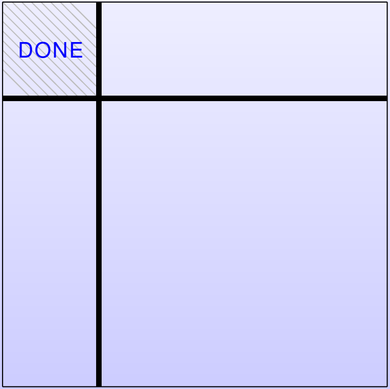
$$L = \left(\begin{array}{c|c} L_{TL} & \\ \hline L_{BL} & L_{BR} \end{array}\right) = ?$$

Dependencies:

$$\left(\begin{array}{c|c} L_{TL} = \Gamma(A_{TL}) & \\ \hline L_{BL} = A_{BL} L_{TL}^{-T} & L_{BR} = \Gamma\big(A_{BR} - L_{BL} L_{BL}^T\big) \end{array}\right)$$

Algorithm #1

Iteration i: completed



Algorithm #1

State of the matrix:

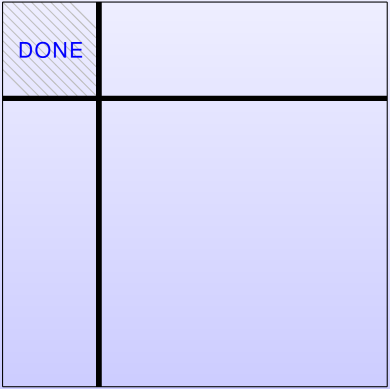
$$\left(\begin{array}{c|c} L_{TL} = \mathrm{CHOL} & \\ \hline & \end{array}\right)$$

Final state:

$$\left(\begin{array}{c|c} L_{TL} = \mathrm{CHOL} & \\ \hline L_{BL} = \mathrm{TRSM} & L_{BR} = \mathrm{CHOL}(\mathrm{SYRK}) \end{array}\right)$$

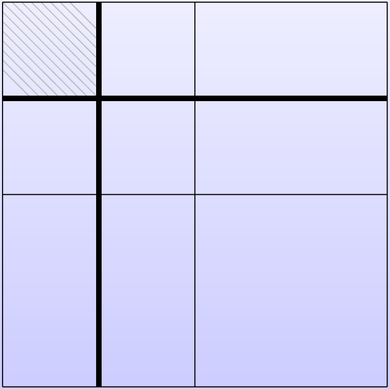
Algorithm #1

Iteration i: completed



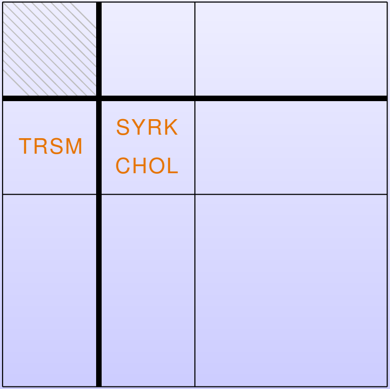
Algorithm #1

Iteration i+1: repartitioning.   Blocked vs. unblocked!



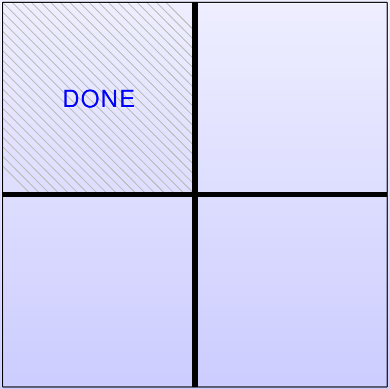
Algorithm #1

Iteration i+1: computation



Algorithm #1

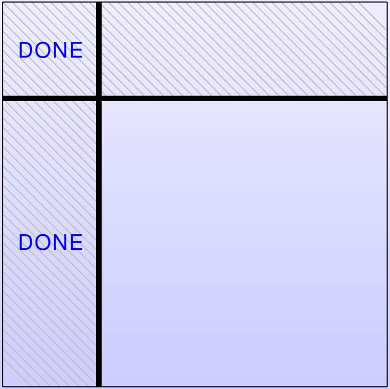
Iteration i+1: completed (boundary shift)



A Different Algorithm?

Algorithm #2

Iteration i: completed



Algorithm #2

State of the matrix:

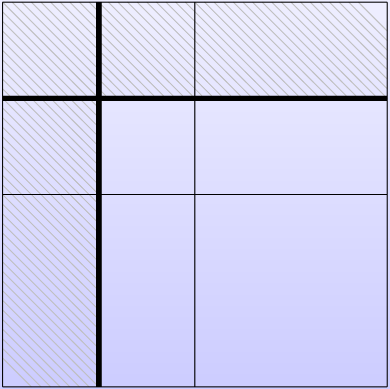
$$\left(\begin{array}{c|c} L_{TL} = \mathrm{CHOL} & \\ \hline L_{BL} = \mathrm{TRSM} & \end{array}\right)$$

Final State:

$$\left(\begin{array}{c|c} L_{TL} = \mathrm{CHOL} & \\ \hline L_{BL} = \mathrm{TRSM} & L_{BR} = \mathrm{CHOL(SYRK)} \end{array}\right)$$

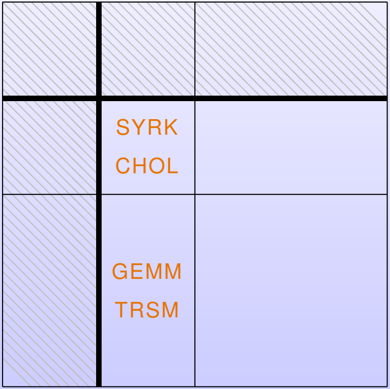
Algorithm #2

Iteration i+1: repartitioning



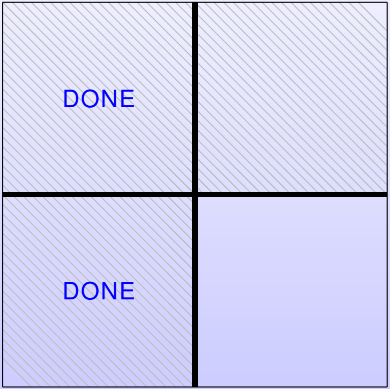
Algorithm #2

Iteration i+1: computation



Algorithm #2

Iteration i+1: completed (boundary shift)



Yet Another Algorithm!

Algorithm #3

State of the matrix:

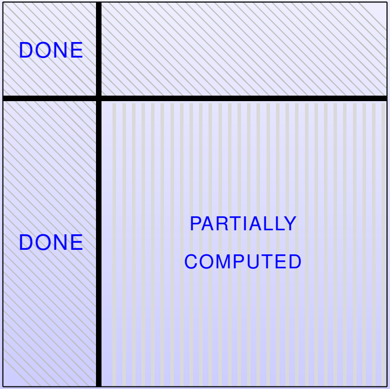
$$\left(\begin{array}{c|c} L_{TL} = \text{CHOL} & \\ \hline L_{BL} = \text{TRSM} & L_{BR} = \text{SYRK} \end{array}\right)$$

Final state:

$$\left(\begin{array}{c|c} L_{TL} = \text{CHOL} & \\ \hline L_{BL} = \text{TRSM} & L_{BR} = \text{CHOL(SYRK)} \end{array}\right)$$

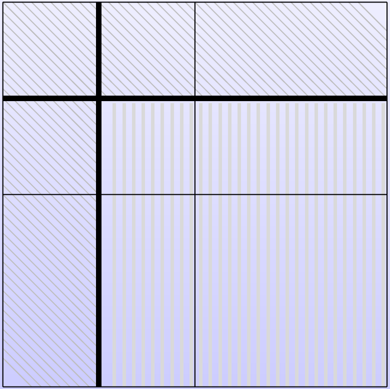
Algorithm #3

Iteration i: completed



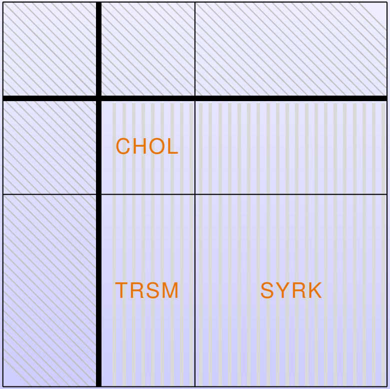
Algorithm #3

Iteration i+1: repartitioning



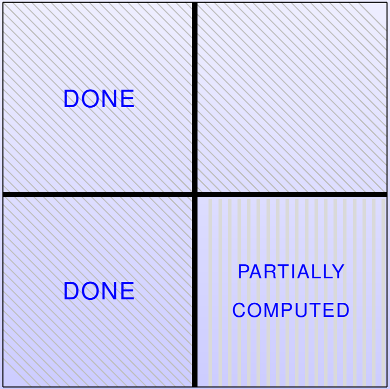
Algorithm #3

Iteration i+1: computation

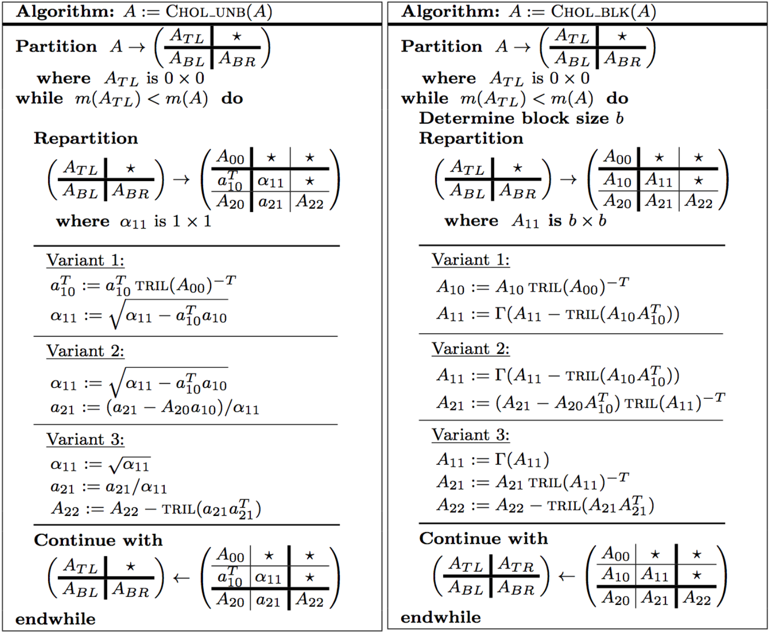


Algorithm #3

Iteration i+1: completed (boundary shift)



Algorithms

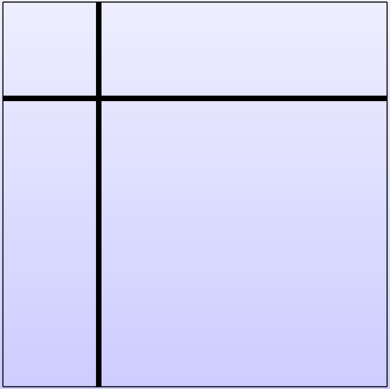


```
Algorithm: $A := \text{CHOL\_UNB}(A)$
Partition $A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$
  where $A_{TL}$ is $0 \times 0$
while $m(A_{TL}) < m(A)$ do

  Repartition
    $\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline a_{10}^T & \alpha_{11} & \star \\ \hline A_{20} & a_{21} & A_{22} \end{array}\right)$
    where $\alpha_{11}$ is $1 \times 1$

    Variant 1:
    $a_{10}^T := a_{10}^T \,\text{TRIL}(A_{00})^{-T}$
    $\alpha_{11} := \sqrt{\alpha_{11} - a_{10}^T a_{10}}$

    Variant 2:
    $\alpha_{11} := \sqrt{\alpha_{11} - a_{10}^T a_{10}}$
    $a_{21} := (a_{21} - A_{20} a_{10}) / \alpha_{11}$

    Variant 3:
    $\alpha_{11} := \sqrt{\alpha_{11}}$
    $a_{21} := a_{21} / \alpha_{11}$
    $A_{22} := A_{22} - \text{TRIL}(a_{21} a_{21}^T)$

  Continue with
    $\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline a_{10}^T & \alpha_{11} & \star \\ \hline A_{20} & a_{21} & A_{22} \end{array}\right)$
endwhile
```

```
Algorithm: $A := \text{CHOL\_BLK}(A)$
Partition $A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$
  where $A_{TL}$ is $0 \times 0$
while $m(A_{TL}) < m(A)$ do
  Determine block size $b$
  Repartition
    $\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$
    where $A_{11}$ is $b \times b$

    Variant 1:
    $A_{10} := A_{10} \,\text{TRIL}(A_{00})^{-T}$
    $A_{11} := \Gamma(A_{11} - \text{TRIL}(A_{10} A_{10}^T))$

    Variant 2:
    $A_{11} := \Gamma(A_{11} - \text{TRIL}(A_{10} A_{10}^T))$
    $A_{21} := (A_{21} - A_{20} A_{10}^T) \,\text{TRIL}(A_{11})^{-T}$

    Variant 3:
    $A_{11} := \Gamma(A_{11})$
    $A_{21} := A_{21} \,\text{TRIL}(A_{11})^{-T}$
    $A_{22} := A_{22} - \text{TRIL}(A_{21} A_{21}^T)$

  Continue with
    $\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$
endwhile
```

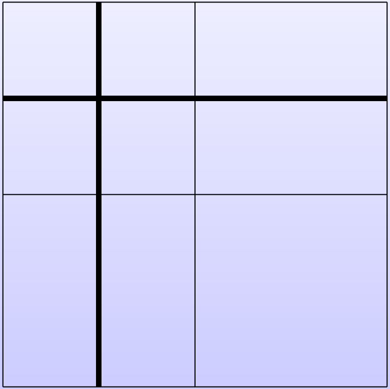
Algorithm Progression

Iteration i: completed



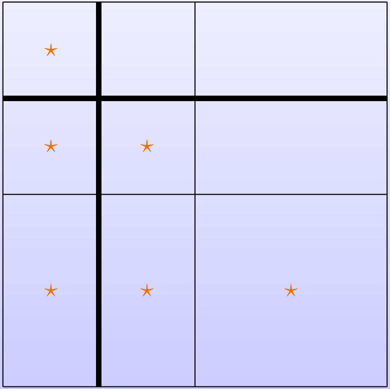
Algorithm Progression

Iteration i+1: repartitioning



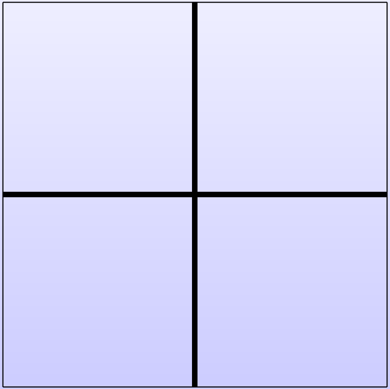
Algorithm Progression

Iteration i+1: computation



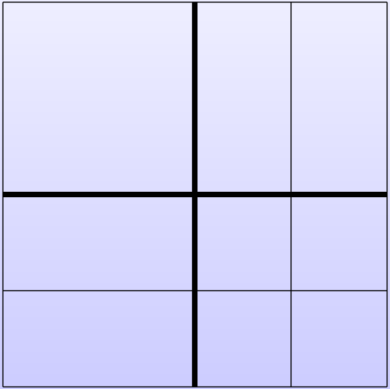
Algorithm Progression

Iteration i+1: completed (boundary shift)



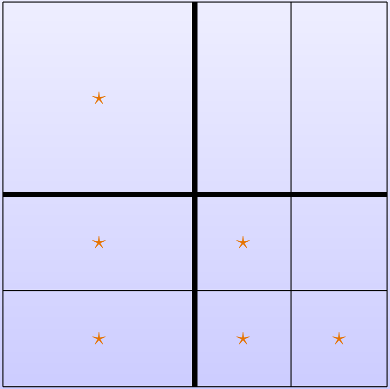
Algorithm Progression

Iteration i+2: repartitioning



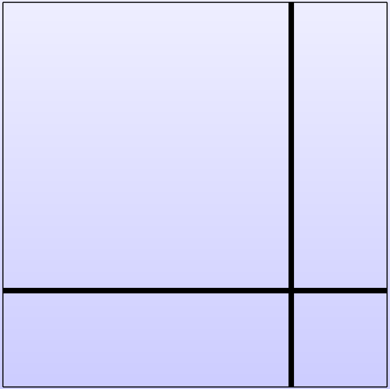
Algorithm Progression

Iteration i+2: computation



Algorithm Progression

Iteration i+2: complete (boundary shift)



Traditional code

- C, triple loop, unblocked.

```
for ( j = 0; j < n; j++ )
{
  A[j,j] = sqrt( A[j,j] );

  for ( i = j+1; i < n; i++ )
    A[i,j] = A[i,j] / A[j,j];

  for ( k = j+1; k < n; k++ )
    for ( i = k; i < n; i++ )
      A[i,k] = A[i,k] - A[i,j] * A[k,j];
}
```

Traditional code

- Matlab, blocked.

```
for j = 1:nb:n,
  b = min( n-j+1, nb );

  A(j:j+b-1, j:j+b-1) = Chol( A(j:j+b-1, j:j+b-1) );

  A(j+b:n,   j:j+b-1) = A(j+b:n, j:j+b-1)/A(j:j+b-1, j:j+b-1)’;

  A(j+b:n,   j+b:n  ) = A(j+b:n, j+b:n) -
                        tril(A(j+b:n, j:j+b-1)) A(j+b:n, j:j+b-1)’;
end
```

Traditional code: LAPACK, blocked

```
SUBROUTINE DPOTRF( UPLO, N, A, LDA, INFO )
[..]
      DO 20 J = 1, N, NB
*
         JB = MIN( NB, N-J+1 )
         CALL DSYRK( 'Lower', 'No transpose', JB, J-1, -ONE,
   $                     A( J, 1 ), LDA, ONE, A( J, J ), LDA )
         CALL DPOTF2( 'Lower', JB, A( J, J ), LDA, INFO )
         IF( INFO.NE.0 )
   $        GO TO 30
         IF( J+JB.LE.N-1 ) THEN
*
            CALL DGEMM( 'No transpose', 'Transpose', N-J-JB+1, JB,
   $                        J-1, -ONE, A( J+JB, 1 ), LDA, A( J, 1 ),
   $                        LDA, ONE, A( J+JB, J ), LDA )
            CALL DTRSM( 'Right', 'Lower', 'Transpose', 'Non-unit',
   $                        N-J-JB+1, JB, ONE, A( J, J ), LDA,
   $                        A( J+JB, J ), LDA )
         END IF
 20   CONTINUE
```

FLAME notation & code

Partition

$$A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

where A_{TL} is 0×0

While $m(A_{TL}) < m(A)$ **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A_{11} is $b \times b$

$$A_{11} := \Gamma(A_{11})$$
$$A_{21} := A_{21}\,\text{TRIL}(A_{11})^{-T}$$
$$A_{22} := A_{22} - \text{TRIL}(A_{21}A_{21}^T)$$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

```
function [ A_out ] = Chol_blk( A, nb_alg )
  [ ATL, ATR, ...
    ABL, ABR ] = FLA_Part_2x2( A, ...
                               0, 0, ’FLA_TL’ );

  while ( size( ATL, 1 ) < size( A, 1 ) )
    b = min( size( ABR, 1 ), nb_alg );

    [ A00, A01, A02, ...
      A10, A11, A12, ...
      A20, A21, A22 ] = FLA_Repart_2x2_to_3x3( ATL, ATR, ...
                                               ABL, ABR, ...
                                               b, b, ’FLA_BR’ );
    % ------------------------------------------------------------%
     A11 = Chol_unb( A11 );
     A21 = A21 / tril( A11 )’;
     A22 = A22 - tril( A21 * A21’ );
    %-------------------------------------------------------------%
    [ ATL, ATR, ...
      ABL, ABR ] = FLA_Cont_with_3x3_to_2x2( A00, A01, A02, ...
                                             A10, A11, A12, ...
                                             A20, A21, A22, ...
                                             ’FLA_TL’ );
  end
  A_out = [ ATL, ATR
            ABL, ABR ];
return
```

FLAME notation & code

Partition

$$A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

where A_{TL} is 0×0

While $m(A_{TL}) < m(A)$ **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A_{11} is $b \times b$

$$\begin{aligned} A_{11} &:= \Gamma(A_{11}) \\ A_{21} &:= A_{21}\,\mathsf{TRIL}(A_{11})^{-T} \\ A_{22} &:= A_{22} - \mathsf{TRIL}(A_{21} A_{21}^T) \end{aligned}$$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

```
function [ A_out ] = Chol_blk( A, nb_alg )
  [ ATL, ATR, ...
    ABL, ABR ] = FLA_Part_2x2( A, ...
                               0, 0, ’FLA_TL’ );

  while ( size( ATL, 1 ) < size( A, 1 ) )
    b = min( size( ABR, 1 ), nb_alg );

    [ A00, A01, A02, ...
      A10, A11, A12, ...
      A20, A21, A22 ] = FLA_Repart_2x2_to_3x3( ATL, ATR, ...
                                                ABL, ABR, ...
                                                b, b, ’FLA_BR’ );
    % ------------------------------------------------------------%
     A11 = Chol_unb( A11 );
     A21 = A21 / tril( A11 )’;
     A22 = A22 - tril( A21 * A21’ );
    %-------------------------------------------------------------%

    [ ATL, ATR, ...
      ABL, ABR ] = FLA_Cont_with_3x3_to_2x2( A00, A01, A02, ...
                                             A10, A11, A12, ...
                                             A20, A21, A22, ...
                                             ’FLA_TL’ );

  end
  A_out = [ ATL, ATR
            ABL, ABR ];
return
```

FLAME notation & code

Partition

$$A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

where A_{TL} is 0×0

While $m(A_{TL}) < m(A)$ **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A_{11} is $b \times b$

$$\begin{aligned} A_{11} &:= \Gamma(A_{11}) \\ A_{21} &:= A_{21}\,\text{TRIL}(A_{11})^{-T} \\ A_{22} &:= A_{22} - \text{TRIL}(A_{21}A_{21}^T) \end{aligned}$$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

```
function [ A_out ] = Chol_blk( A, nb_alg )
  [ ATL, ATR, ...
    ABL, ABR ] = FLA_Part_2x2( A, ...
                               0, 0, 'FLA_TL' );

  while ( size( ATL, 1 ) < size( A, 1 ) )
    b = min( size( ABR, 1 ), nb_alg );

    [ A00, A01, A02, ...
      A10, A11, A12, ...
      A20, A21, A22 ] = FLA_Repart_2x2_to_3x3( ATL, ATR, ...
                                               ABL, ABR, ...
                                               b, b, 'FLA_BR' );
    % ------------------------------------------------------------%
     A11 = Chol_unb( A11 );
     A21 = A21 / tril( A11 )';
     A22 = A22 - tril( A21 * A21' );
    %------------------------------------------------------------%
    [ ATL, ATR, ...
      ABL, ABR ] = FLA_Cont_with_3x3_to_2x2( A00, A01, A02, ...
                                             A10, A11, A12, ...
                                             A20, A21, A22, ...
                                             'FLA_TL' );
  end
  A_out = [ ATL, ATR
            ABL, ABR ];
return
```

FLAME notation & code

Partition

$$A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

where A_{TL} is 0×0

While $m(A_{TL}) < m(A)$ **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A_{11} is $b \times b$

$$A_{11} := \Gamma(A_{11})$$
$$A_{21} := A_{21}\ \mathsf{TRIL}(A_{11})^{-T}$$
$$A_{22} := A_{22} - \mathsf{TRIL}(A_{21} A_{21}^T)$$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

```
function [ A_out ] = Chol_blk( A, nb_alg )
  [ ATL, ATR, ...
    ABL, ABR ] = FLA_Part_2x2( A, ...
                               0, 0, ’FLA_TL’ );

  while ( size( ATL, 1 ) < size( A, 1 ) )
    b = min( size( ABR, 1 ), nb_alg );
    [ A00, A01, A02, ...
      A10, A11, A12, ...
      A20, A21, A22 ] = FLA_Repart_2x2_to_3x3( ATL, ATR, ...
                                               ABL, ABR, ...
                                               b, b, ’FLA_BR’ );
    % -----------------------------------------------------------%
     A11 = Chol_unb( A11 );
     A21 = A21 / tril( A11 )’;
     A22 = A22 - tril( A21 * A21’ );
    %------------------------------------------------------------%

    [ ATL, ATR, ...
      ABL, ABR ] = FLA_Cont_with_3x3_to_2x2( A00, A01, A02, ...
                                             A10, A11, A12, ...
                                             A20, A21, A22, ...
                                             ’FLA_TL’ );
  end
  A_out = [ ATL, ATR
            ABL, ABR ];
return
```

FLAME notation & code

Partition

$$A \rightarrow \left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right)$$

where A_{TL} is 0×0

While $m(A_{TL}) < m(A)$ **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A_{11} is $b \times b$

$$A_{11} := \Gamma(A_{11})$$
$$A_{21} := A_{21}\ \mathrm{TRIL}(A_{11})^{-T}$$
$$A_{22} := A_{22} - \mathrm{TRIL}(A_{21} A_{21}^T)$$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & \star \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & \star & \star \\ \hline A_{10} & A_{11} & \star \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

```
function [ A_out ] = Chol_blk( A, nb_alg )

  [ ATL, ATR, ...
    ABL, ABR ] = FLA_Part_2x2( A, ...
                               0, 0, 'FLA_TL' );

  while ( size( ATL, 1 ) < size( A, 1 ) )
    b = min( size( ABR, 1 ), nb_alg );

    [ A00, A01, A02, ...
      A10, A11, A12, ...
      A20, A21, A22 ] = FLA_Repart_2x2_to_3x3( ATL, ATR, ...
                                                ABL, ABR, ...
                                                b, b, 'FLA_BR' );
    % ------------------------------------------------------------%
     A11 = Chol_unb( A11 );
     A21 = A21 / tril( A11 )';
     A22 = A22 - tril( A21 * A21' );
    %-------------------------------------------------------------%

    [ ATL, ATR, ...
      ABL, ABR ] = FLA_Cont_with_3x3_to_2x2( A00, A01, A02, ...
                                             A10, A11, A12, ...
                                             A20, A21, A22, ...
                                             'FLA_TL' );
  end
  A_out = [ ATL, ATR
            ABL, ABR ];
return
```