Goal-oriented and Modular Stability Analysis

Paolo Bientinesi

AICES, RWTH Aachen
pauldj@aices.rwth-aachen.de

Conference on Numerical Linear Algebra:
Perturbation, Performance and Portability
July 19-21, 2010
Austin, TX

Sylvester Equation: historical note

$$AX + XB = C$$

- **R.H. Bartels, G.W. Stewart**. *Solution of the matrix equation AX + XB = C* Communications of the ACM, **1972**

Sylvester Equation: historical note

$$AX + XB = C$$

- **R.H. Bartels, G.W. Stewart**. *Solution of the matrix equation AX + XB = C* Communications of the ACM, **1972**
- B. Kågström, P. Poromaa. *Distributed and shared memory block algorithms for the triangular Sylvester equation [..]* SIMAX, 1992
- M. Marqués, R. Van de Geijn and V. Hernández. *Parallel algorithms for the triangular Sylvester equation.* 3rd SIAM Conference [..] 1993
- B. Kågström, P. Poromaa. *LAPACK-style algorithms and software for solving the generalized Sylvester equation [..]* TOMS, 1996
- I. Jonsson, B. Kågström. *Recursive blocked algorithms [..]. Part I: one-sided and coupled Sylvester [..]* TOMS, 2002
- R. Granat, B. Kagstrom, and P. Poromaa. *Parallel ScaLAPACK-style algorithms for solving continuous-time Sylvester [..]* EuroPar 2003

Sylvester Equation: 2001 circa

- FLAME project
- E. Quintana-Ortí, R. van de Geijn: 16 algorithms

$$AX + XB = C \quad \equiv \quad X = \Omega(A, B, C)$$

Partition $\star \in \{A, B, C\}$ **as** $\left(\begin{array}{c|c} \star_{TL} & \star_{TR} \\ \hline \star_{BL} & \star_{BR} \end{array}\right)$ **where** A_{BR}, B_{TL}, C_{BL} are 0×0

While $size(C_{TL}) < size(C)$ **do**

Repartition

$$\left(\begin{array}{c|c} C_{TL} & C_{TR} \\ \hline C_{BL} & C_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} C_{00} & C_{01} & C_{02} \\ \hline C_{10} & C_{11} & C_{12} \\ \hline C_{20} & C_{21} & C_{22} \end{array}\right), \left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right), \ldots$$

Algorithm 1	...	Algorithm 16
$C_{10} := C_{10} - A_{12} C_{20}$	$C_{10} := C_{10} - A_{12} C_{20}$	$C_{11} := C_{11} - C_{10} B_{01}$
$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$
$C_{21} := C_{21} - C_{20} B_{01}$	$C_{11} := C_{11} - C_{10} B_{01} - A_{12} C_{21}$	$C_{01} := C_{01} - C_{00} B_{01} - A_{01} C_{11}$
$C_{21} := \Omega(A_{22}, B_{11}, C_{21})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{01} := \Omega(A_{00}, B_{11}, C_{10})$
$C_{11} := C_{11} - A_{12} C_{21} - C_{10} B_{01}$	$C_{12} := C_{12} - C_{10} B_{02} - C_{11} B_{12}$	$C_{12} := C_{12} - C_{10} B_{02} - C_{11} B_{12}$
$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{12} := C_{12} - A_{12} C_{22}$	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$
	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$	$C_{02} := C_{02} - A_{01} C_{12}$

Continue

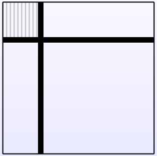
$$\left(\begin{array}{c|c} C_{TL} & C_{TR} \\ \hline C_{BL} & C_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} C_{00} & C_{01} & C_{02} \\ \hline C_{10} & C_{11} & C_{12} \\ \hline C_{20} & C_{21} & C_{22} \end{array}\right), \left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right), \ldots$$

endwhile

Partition

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

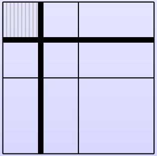
where A_{TL} **is empty**



While A_{TL} <> A

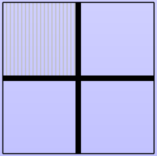
Repartition

$$\left\{ \left(\begin{array}{cc} A_{TL} & A_{TR} \\ A_{BL} & A_{BR} \end{array} \right) \rightarrow \left(\begin{array}{c|cc} A_{00} & \left(A_{01} \right. & \left. A_{02} \right) \\ \hline \left(\begin{array}{c} A_{10} \\ A_{20} \end{array} \right) & \left(\begin{array}{c} A_{11} \\ A_{21} \end{array} \right. & \left. \begin{array}{c} A_{12} \\ A_{22} \end{array} \right) \end{array} \right) \right\}$$



Continue with

$$\left\{ \left(\begin{array}{cc|c} \left(\begin{array}{c} A_{00} \\ A_{10} \end{array} \right. & \left. \begin{array}{c} A_{01} \\ A_{11} \end{array} \right) & \left(\begin{array}{c} A_{02} \\ A_{12} \end{array} \right) \\ \hline \left(A_{20} \right. & \left. A_{21} \right) & A_{22} \end{array} \right) \rightarrow \left(\begin{array}{cc} A_{TL} & A_{TR} \\ A_{BL} & A_{BR} \end{array} \right) \right\}$$



end while

Back to the 16 algorithms

Algorithm 1	...	Algorithm 16
$C_{10} := C_{10} - A_{12}C_{20}$	$C_{10} := C_{10} - A_{12}C_{20}$	$C_{11} := C_{11} - C_{10}B_{01}$
$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$
$C_{21} := C_{21} - C_{20}B_{01}$	$C_{11} := C_{11} - C_{10}B_{01} - A_{12}C_{21}$	$C_{01} := C_{01} - C_{00}B_{01} - A_{01}C_{11}$
$C_{21} := \Omega(A_{22}, B_{11}, C_{21})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{01} := \Omega(A_{00}, B_{11}, C_{10})$
$C_{11} := C_{11} - A_{12}C_{21} - C_{10}B_{01}$	$C_{12} := C_{12} - C_{10}B_{02} - C_{11}B_{12}$	$C_{12} := C_{12} - C_{10}B_{02} - C_{11}B_{12}$
$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{12} := C_{12} - A_{12}C_{22}$	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$
	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$	$C_{02} := C_{02} - A_{01}C_{12}$

Good performance → TOMS submission

Back to the 16 algorithms

Algorithm 1	...	Algorithm 16
$C_{10} := C_{10} - A_{12}C_{20}$	$C_{10} := C_{10} - A_{12}C_{20}$	$C_{11} := C_{11} - C_{10}B_{01}$
$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{10} := \Omega(A_{11}, B_{00}, C_{10})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$
$C_{21} := C_{21} - C_{20}B_{01}$	$C_{11} := C_{11} - C_{10}B_{01} - A_{12}C_{21}$	$C_{01} := C_{01} - C_{00}B_{01} - A_{01}C_{11}$
$C_{21} := \Omega(A_{22}, B_{11}, C_{21})$	$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{01} := \Omega(A_{00}, B_{11}, C_{10})$
$C_{11} := C_{11} - A_{12}C_{21} - C_{10}B_{01}$	$C_{12} := C_{12} - C_{10}B_{02} - C_{11}B_{12}$	$C_{12} := C_{12} - C_{10}B_{02} - C_{11}B_{12}$
$C_{11} := \Omega(A_{11}, B_{11}, C_{11})$	$C_{12} := C_{12} - A_{12}C_{22}$	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$
	$C_{12} := \Omega(A_{11}, B_{22}, C_{12})$	$C_{02} := C_{02} - A_{01}C_{12}$

Good performance → TOMS submission

Problem: missing error analysis

- Are they all numerically stable? (no!)
- Recursive calls? k^2 stability analyses? hm...
- **Initial Goal:** (dissertation)
  Systematic procedure from algorithm to stability?

  Systematic ⇒ computer algebra system

Detour: symbolic system

Automatic generation of algorithms

$$\{A, L, U \in \mathbb{R}^{n \times n}$$
$$\texttt{LowerTriUni}(L),$$
$$\texttt{UpperTri}(U),$$
$$LU = A\ \} \quad \Rightarrow$$

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

where A_{TL} is 0×0

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

Repartition

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ A_{20} & A_{21} & A_{22} \end{array}\right)$$

$$A_{01} := L_{00}^{-1} A_{01}$$
$$A_{10} := A_{10} U_{00}^{-1}$$
$$A_{11} := \mathrm{LU}(A_{11} - A_{10} A_{01})$$

Continue

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

Not a detour afterall

Stability analysis as a matrix operation

- **Operation:** $LU = A$
  Input: A
  Goal: find algorithm(s) to compute L and U

RWTH AACHEN UNIVERSITY

Stability analysis as a matrix operation

- **Operation:** $LU = A$
  Input: A
  Goal: find algorithm(s) to compute L and U

- **Analysis:** $\check{L}\check{U} = A + \Delta A$
  Input: $A, \check{L}, \check{U}$

AICES

Stability analysis as a matrix operation

- **Operation:** $LU = A$
  Input: A
  Goal: find algorithm(s) to compute L and U

- **Analysis:** $\check{L}\check{U} = A + \Delta A$
  Input: $A, \check{L}, \check{U}$
  Input: algorithm used to compute L and U

Stability analysis as a matrix operation

- **Operation:** $LU = A$
  Input: A
  Goal: find algorithm(s) to compute L and U

- **Analysis:** $\check{L}\check{U} = A + \Delta A$
  Input: $A, \check{L}, \check{U}$
  Input: algorithm used to compute L and U
  Goal: find algorithm(s) to "compute" ΔA

Not a detour afterall

Stability analysis as a matrix operation

- **Operation:** $LU = A$
  Input: A
  Goal: find algorithm(s) to compute L and U

- **Analysis:** $\check{L}\check{U} = A + \Delta A$
  Input: $A, \check{L}, \check{U}$
  Input: algorithm used to compute L and U
  Goal: find algorithm(s) to "compute" ΔA
  Goal: find bounds for $\|\Delta A\|$ or $|\Delta A|$

The methodology for generating algorithms can be used for studying their numerical properties too.

Definition: loop-invariant

Example: Sorting

$$\texttt{Sort(}\ v = \{x_0, x_1, x_2, x_3, \ldots, x_n\}\ \texttt{)}$$

$$\texttt{Sort(}\ \ v = \{x_0, x_1, x_2, x_3, \ldots, x_n\}\ \ \texttt{)}$$

Algorithm: selection sort

```
for i=0:n,
   swap( min(v(i:n)), v(i) );
endfor
```

$$\texttt{Sort(}\ v = \{x_0, x_1, x_2, x_3, \ldots, x_n\}\ \texttt{)}$$

Algorithm: selection sort

```
for i=0:n,
   swap( min(v(i:n)), v(i) );
endfor
```

After `i`-th iteration:

$$\{\underbrace{v_0, v_1, \ldots, v_i}_{v_L = \text{ sorted}} \ \mid\ \underbrace{v_{i+1}, v_{i+2}, \ldots, v_n}_{v_R = \text{to sort}}\}$$

$$\texttt{Sort(}\ v = \{x_0, x_1, x_2, x_3, \ldots, x_n\}\ \texttt{)}$$

Algorithm: selection sort

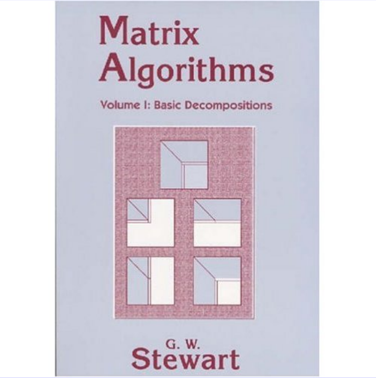
```
for i=0:n,
   swap( min(v(i:n)), v(i) );
endfor
```

After `i`-th iteration:

$$\underbrace{\{v_0, v_1, \ldots, v_i}_{v_L = \text{ sorted}} \;|\; \underbrace{v_{i+1}, v_{i+2}, \ldots, v_n\}}_{v_R = \text{to sort}}$$

Loop Invariant: $v = \{\, v_L \,|\, v_R \,\} \;\wedge\; sorted(v_L)$

Bordered variant



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

where A_{TL} is 0×0

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

Repartition

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

$$A_{01} := L_{00}^{-1} A_{01}$$
$$A_{10} := A_{10} U_{00}^{-1}$$
$$A_{11} := \mathrm{LU}(A_{11} - A_{10} A_{01})$$

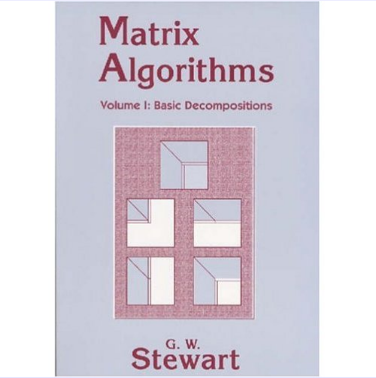
Continue

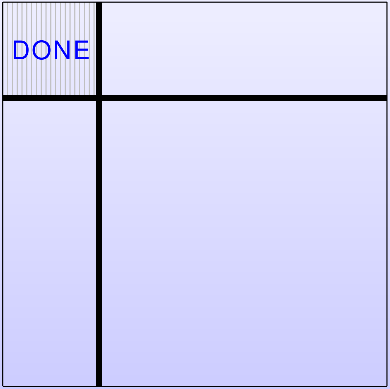
$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

Example: LU factorization

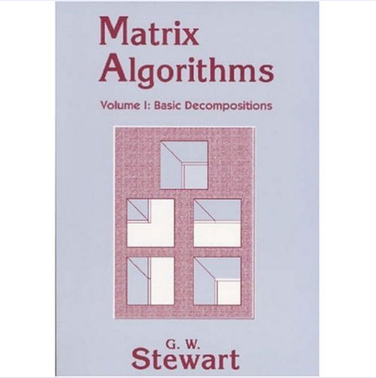
Iteration i: completed

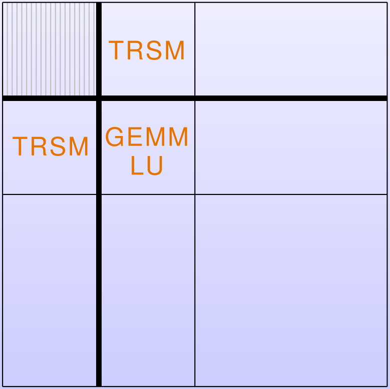




Example: LU factorization

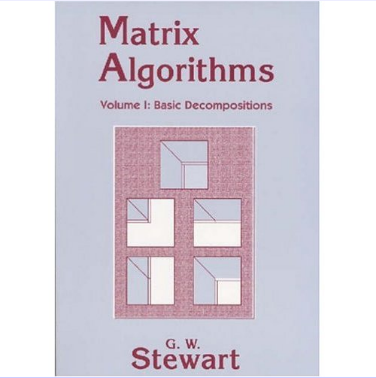
Iteration i+1: computation

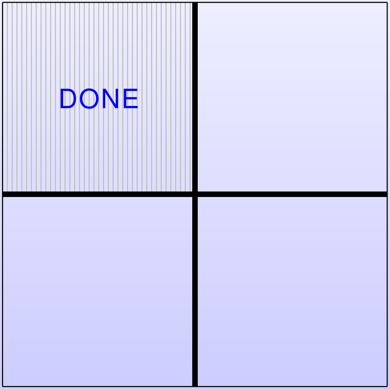




Example: LU factorization

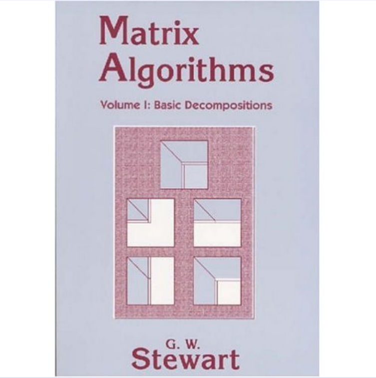
Iteration i+1: completed (boundary shift)



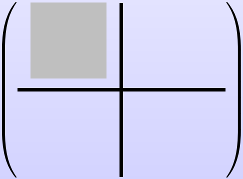


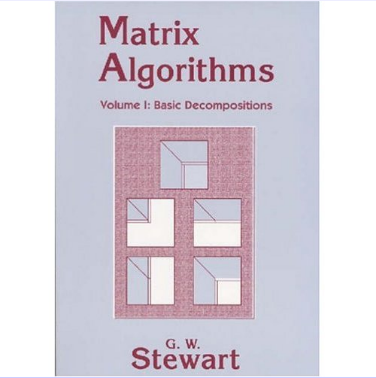
Example: LU factorization

Loop-Invariant



Loop-Invariant = ?



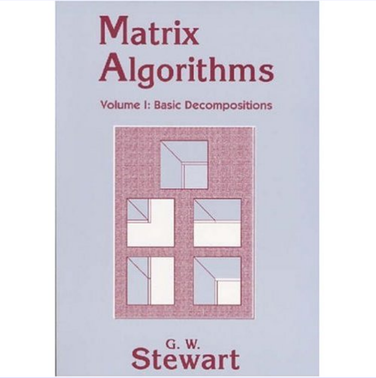


PME$(LU = A)$

$$\left(\begin{array}{c|c} L_{TL}U_{TL} = A_{TL} & U_{TR} = L_{TL}^{-1}A_{TR} \\ \hline L_{BL} = A_{BL}U_{TL}^{-1} & \begin{array}{l} L_{BR}U_{BR} = \\ A_{BR} - L_{BL}U_{TR} \end{array} \end{array}\right)$$

Example: LU factorization

Loop-Invariant

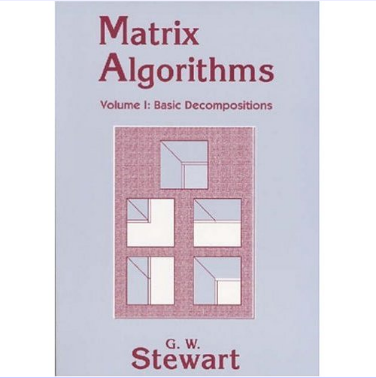


Loop-Invariant

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

Example: LU factorization

Crout variant



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

where A_{TL} is 0×0

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

Repartition

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

$$A_{11} := \mathrm{LU}(A_{11} - A_{10}A_{01})$$
$$A_{12} := (A_{21} - A_{20}A_{01})U_{11}^{-1}$$
$$A_{21} := L_{11}^{-1}(A_{12} - A_{10}A_{02})$$

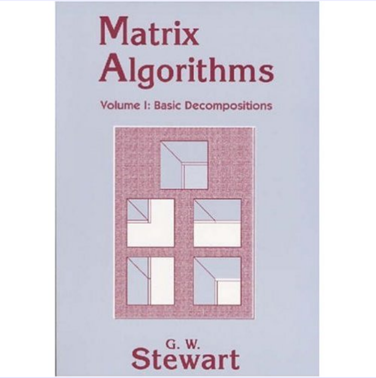
Continue

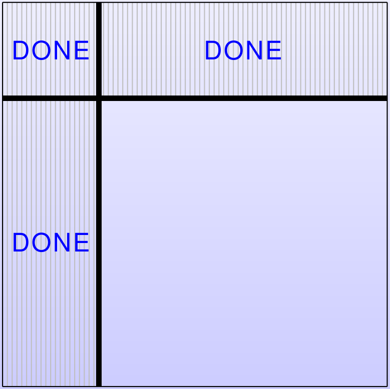
$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

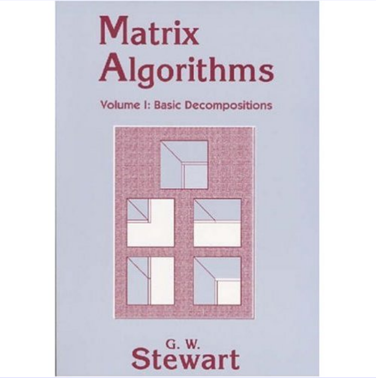
Example: LU factorization

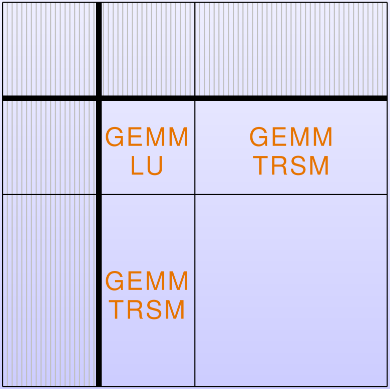
Iteration i: completed





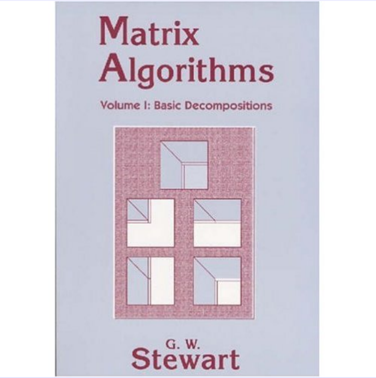
Iteration i+1: computation

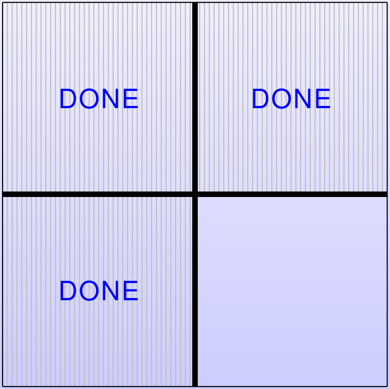


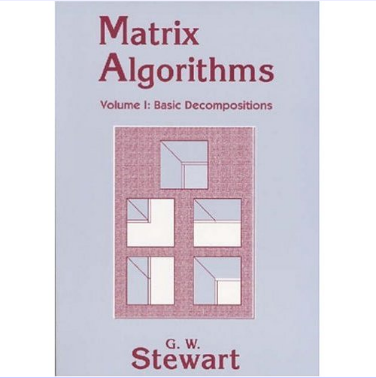


Example: LU factorization

Iteration i+1: completed (boundary shift)





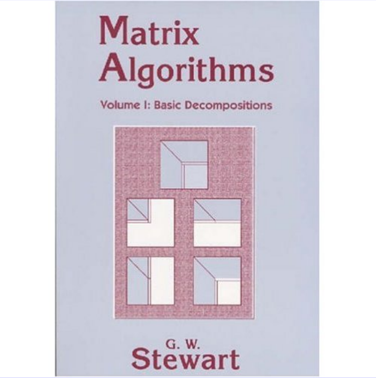


PME$(LU = A)$

$$\left(\begin{array}{c|c} L_{TL}U_{TL} = A_{TL} & U_{TR} = L_{TL}^{-1}A_{TR} \\ \hline L_{BL} = A_{BL}U_{TL}^{-1} & \begin{array}{l} L_{BR}U_{BR} = \\ A_{BR} - L_{BL}U_{TR} \end{array} \end{array}\right)$$

Example: LU factorization

Loop-Invariant



Loop-Invariant

$$\left(\begin{array}{c|c} L_{TL}U_{TL} = A_{TL} & U_{TR} = L_{TL}^{-1}A_{TR} \\ \hline L_{BL} = A_{BL}U_{TL}^{-1} & \end{array}\right)$$

RWTH AACHEN UNIVERSITY

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

where A_{TL} is 0×0

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

Repartition

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ \hline A_{10} & A_{11} & A_{12} \\ A_{20} & A_{21} & A_{22} \end{array}\right)$$

$A_{01} := L_{00}^{-1} A_{01}$

$A_{10} := A_{10} U_{00}^{-1}$

$A_{11} := \mathrm{LU}(A_{11} - A_{10} A_{01})$

Continue

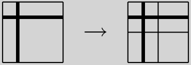
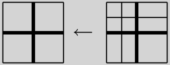
$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & A_{01} & A_{02} \\ A_{10} & A_{11} & A_{12} \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

Step	
1a	$\{$ Precondition $\}$
4	**Partition** . . . **where** . . .
2	$\{$ Loop-Inv $\}$
3	**While** G **do**
2,3	$\{($ Loop-Inv $) \wedge ($ G $)\}$
5a	**Repartition** . . . **where** . . .
6	$\{$ LI-Before $\}$
8	*Updates*
7	$\{$ LI-After $\}$
5b	**Continue with** . . .
2	$\{$ Loop-Inv $\}$
	endwhile
2,3	$\{($ Loop-Inv $) \wedge \neg ($ G $)\}$
1b	$\{$ Postcondition $\}$

AICES

Worksheet: LU

Step	LU
1a	$\{ \; A, L, U \in \mathbb{R}^{n \times n}, \quad \texttt{LowerTriUni}(L), \quad \texttt{UpperTri}(U) \; \}$
4	**Partition** A, L, U **where ...**
2	$\{ \; L_{TL} U_{TL} = A_{TL} \; \}$
3	**While** $size(A_{TL}) < size(A)$ **do**
2,3	$\{(\; L_{TL} U_{TL} = A_{TL} \;) \wedge (\; size(A_{TL}) < size(A) \;)\}$
5a	**Repartition**  **where ...**
6	$\{ \; \text{LI-Before} \; \}$
8	$A_{01} := L_{00}^{-1} A_{01}$ $A_{10} := A_{10} U_{00}^{-1}$ $A_{11} := \mathsf{LU}(A_{11} - A_{10} A_{01})$
7	$\{ \; \text{LI-After} \; \}$
5b	**Continue with** 
2	$\{ \; L_{TL} U_{TL} = A_{TL} \; \}$
	endwhile
2,3	$\{(\; L_{TL} U_{TL} = A_{TL} \;) \wedge \neg (\; size(A_{TL}) < size(A) \;)\}$
1b	$\{ \; LU = A \; \}$

Extended worksheet

$$LU = A \qquad \Rightarrow \qquad \check{L}\check{U} = A + \Delta A$$

Operation	Analysis	Step
Partition Operands, where . . .	Error Operands	4
{ Loop-Invariant }	{ Error-Invariant }	2
While G **do**		3
{ Loop-Invariant }	{ Error-Invariant }	2,3
Repartition Operands, where . . .	Error Operands	5a
{ LI-Before }	{ Err-Before }	6
Updates	*Error Updates*	8
{ LI-After }	{ Err-After }	7
Continue with . . .		5b
{ Loop-Invariant }	{ Error-Invariant }	2
endwhile		

Example: $Lx = b$

Partition $L \rightarrow \left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right), x \rightarrow \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right), b \rightarrow \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right)$

where L_{TL} is 0×0

While $size(L_{TL}) < size(L)$ **do**

Repartition

$$\left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} L_{00} & 0 & 0 \\ \hline l_{10}^T & \lambda_{11} & 0 \\ \hline L_{20} & l_{21} & L_{22} \end{array}\right), \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) \rightarrow \left(\begin{array}{c} x_0 \\ \hline \chi_1 \\ \hline x_2 \end{array}\right), \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right) \rightarrow \left(\begin{array}{c} b_0 \\ \hline \beta_1 \\ \hline b_2 \end{array}\right)$$

$$\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$$

Continue

$$\left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} L_{00} & 0 & 0 \\ \hline l_{10}^T & \lambda_{11} & 0 \\ \hline L_{20} & l_{21} & L_{22} \end{array}\right), \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) \leftarrow \left(\begin{array}{c} x_0 \\ \hline \chi_1 \\ \hline x_2 \end{array}\right), \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right) \leftarrow \left(\begin{array}{c} b_0 \\ \hline \beta_1 \\ \hline b_2 \end{array}\right)$$

endwhile

Questions

- $(L + \Delta L)\check{x} = b$? $\quad (L + \Delta L)\check{x} = (b + \delta b)$?
- $|\Delta L| \leq O(n)\mathbf{u}|L|$?

RWTH AACHEN UNIVERSITY

Partition $L \rightarrow \left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right), x \rightarrow \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right), b \rightarrow \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right)$

where L_{TL} is 0×0

While $size(L_{TL}) < size(L)$ **do**

Repartition

$$\left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} L_{00} & 0 & 0 \\ \hline l_{10}^T & \lambda_{11} & 0 \\ \hline L_{20} & l_{21} & L_{22} \end{array}\right), \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) \rightarrow \left(\begin{array}{c} x_0 \\ \hline \chi_1 \\ \hline x_2 \end{array}\right), \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right) \rightarrow \left(\begin{array}{c} b_0 \\ \hline \beta_1 \\ \hline b_2 \end{array}\right)$$

$$\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$$

Continue

$$\left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} L_{00} & 0 & 0 \\ \hline l_{10}^T & \lambda_{11} & 0 \\ \hline L_{20} & l_{21} & L_{22} \end{array}\right), \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) \leftarrow \left(\begin{array}{c} x_0 \\ \hline \chi_1 \\ \hline x_2 \end{array}\right), \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right) \leftarrow \left(\begin{array}{c} b_0 \\ \hline \beta_1 \\ \hline b_2 \end{array}\right)$$

endwhile

Invariants?

- Loop-invariant: $L_{TL} x_T = b_T$
- Error-invariant: $(L_{TL} + \Delta L_{TL}) \check{x}_T = b_T$

AI ces

Extended worksheet: TRSV

<table>
<tr><td>$Lx = b$ $(L + \Delta L)\check{x} = b$</td><td>Step</td></tr>
<tr><td>Partition $L \rightarrow \left(\begin{array}{c|c} L_{TL} & 0 \\ \hline L_{BL} & L_{BR} \end{array}\right), x \rightarrow \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right), b \rightarrow \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right),$ $\Delta L \rightarrow \left(\begin{array}{c|c} \Delta L_{TL} & 0 \\ \hline \Delta L_{BL} & \Delta L_{BR} \end{array}\right)$
where $L_{TL}, \Delta L_{TL}$ are 0×0</td><td>4</td></tr>
<tr><td>$\{ \; (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \; \}$</td><td>2</td></tr>
<tr><td>While $size(L_{TL}) < size(L)$ do</td><td>3</td></tr>
<tr><td>$\{ \; (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \; \}$</td><td>2,3</td></tr>
<tr><td>Repartition
$\ldots, \left(\begin{array}{c|c} \Delta L_{TL} & 0 \\ \hline \Delta L_{BL} & \Delta L_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} \Delta L_{00} & 0 & 0 \\ \hline \delta_{10}^T & \delta_{11} & 0 \\ \Delta L_{20} & \delta_{21} & \Delta L_{22} \end{array}\right)$</td><td>5a</td></tr>
<tr><td>$\{ \; (L_{00} + \Delta L_{00})\check{x}_0 = b_0 \; \}$</td><td>6</td></tr>
<tr><td>$\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$ | *Error Updates*</td><td>8</td></tr>
<tr><td>$\left\{ \left(\begin{array}{c} (L_{00} + \Delta L_{00})\check{x}_0 = b_0 \\ \hline (l_{10}^T + \delta_{10}^T)\,\check{x}_0 + (\lambda_{11} + \delta_{11})\,\check{\chi}_1 = \beta_1 \end{array} \right) \right\}$</td><td>7</td></tr>
<tr><td>Continue with
$\ldots, \left(\begin{array}{c|c} \Delta L_{TL} & 0 \\ \hline \Delta L_{BL} & \Delta L_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} \Delta L_{00} & 0 & 0 \\ \hline \delta_{10}^T & \delta_{11} & 0 \\ \hline \Delta L_{20} & \delta_{21} & \Delta L_{22} \end{array}\right)$</td><td>5b</td></tr>
<tr><td>$\{ \; (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \; \}$</td><td>2</td></tr>
<tr><td>endwhile</td><td></td></tr>
</table>

RWTH AACHEN UNIVERSITY

Current status: $(L_{00} + \Delta L_{00})\check{x}_0 = b_0$

Computation: $\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$

After:

$$\left(\frac{(L_{00} + \Delta L_{00})\check{x}_0 = b_0}{\left(l_{10}^T + \delta l_{10}^T\right)\check{x}_0 + \left(\lambda_{11} + \delta\lambda_{11}\right)\check{\chi}_1 = \beta_1}\right)$$

Sub-goal

Can the error generated by $\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$ be accumulated into δl_{10}^T and $\delta\lambda_{11}$?

AICES

Computational models

- $[\chi \text{ op } \psi] = (\chi \text{ op } \psi)(1+\epsilon), \qquad |\epsilon| \leq \mathbf{u}, \text{ and op } \in \{+,-,*,/\}$
- $[\chi \text{ op } \psi] = \dfrac{\chi \text{ op } \psi}{1+\epsilon}, \qquad |\epsilon| \leq \mathbf{u}, \text{ and op } \in \{+,-,*,/\}$

1: $\lambda\check{\nu} = (\beta + \delta\beta) - (l + \delta l)^T x \qquad \begin{cases} |\delta\beta| \leq \gamma_2|\beta| \\ |\delta l| \leq \gamma_{n+2}|l| \end{cases}$

2: $(\lambda + \delta\lambda)\check{\nu} = \beta - (l + \delta l)^T x \qquad \begin{cases} |\delta\lambda| \leq \gamma_2|\lambda| \\ |\delta l| \leq \gamma_n|l| \end{cases}$

3: $(\lambda + \delta\lambda)\check{\nu} = (\beta + \delta\beta) - (l + \delta l)^T x \begin{cases} |\delta\beta| \leq \gamma_1|\beta| \\ |\delta l| \leq \gamma_{n+1}|l| \\ |\delta\lambda| \leq \gamma_1|\lambda| \end{cases}$

Computational models

- $[\chi \text{ op } \psi] = (\chi \text{ op } \psi)(1+\epsilon), \qquad |\epsilon| \leq \mathbf{u}, \text{ and op } \in \{+,-,*,/\}$
- $[\chi \text{ op } \psi] = \dfrac{\chi \text{ op } \psi}{1+\epsilon}, \qquad |\epsilon| \leq \mathbf{u}, \text{ and op } \in \{+,-,*,/\}$

1: $\lambda\check{\nu} = (\beta + \delta\beta) - (l + \delta l)^T x \qquad \begin{cases} |\delta\beta| \leq \gamma_2|\beta| \\ |\delta l| \leq \gamma_{n+2}|l| \end{cases}$

2: $(\lambda + \delta\lambda)\check{\nu} = \beta - (l + \delta l)^T x \qquad \begin{cases} |\delta\lambda| \leq \gamma_2|\lambda| \\ |\delta l| \leq \gamma_n|l| \end{cases}$

3: $(\lambda + \delta\lambda)\check{\nu} = (\beta + \delta\beta) - (l + \delta l)^T x \begin{cases} |\delta\beta| \leq \gamma_1|\beta| \\ |\delta l| \leq \gamma_{n+1}|l| \\ |\delta\lambda| \leq \gamma_1|\lambda| \end{cases}$

TRSV: result

$Lx = b$ $(L + \Delta L)\check{x} = b$	Step
Partition $L, x, b, \quad \Delta L$	4
$\{ \ (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \ \}$	2
While $size(L_{TL}) < size(L)$ **do**	3
$\{ \ (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \ \}$	2,3
Repartition	5a
$\chi_1 := (\beta_1 - l_{10}^T x_0)/\lambda_{11}$ | $\beta_1 = \left(l_{10}^T + \delta l_{10}^T \right) \check{x}_0 + (\lambda_{11} + \delta\lambda_{11})\, \check{\chi}_1$ $\wedge \left| \left(\delta l_{10}^T \,\middle\|\, \delta\lambda_{11} \right) \right| \leq \max(\gamma_2, \gamma_k) \left| \left(l_{10}^T \,\middle\|\, \lambda_{11} \right) \right|$	8
Continue with	5b
$\{ \ (L_{TL} + \Delta L_{TL})\check{x}_T = b_T \ \}$	2
endwhile	

On exit:

- $(L_{TL} + \Delta L_{TL})\check{x}_T = b_T \quad \wedge \quad size(L_{TL}) = size(L) \implies (L + \Delta L)\check{x} = b$
- $|\Delta L| \leq \max(\gamma_2, \gamma_{n-1})\, |L|$

$LU = A$ Crout variant

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

where A_{TL} is 0×0

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

Repartition

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & a_{01} & A_{02} \\ \hline a_{10}^T & \alpha_{11} & a_{12}^T \\ \hline A_{20} & a_{21} & A_{22} \end{array}\right)$$

$$\alpha_{11} := \alpha_{11} - a_{10}^T a_{01}$$
$$a_{21} := (a_{21} - A_{20} a_{01}) / \alpha_{11}$$
$$a_{12}^T := (a_{12}^T - a_{10}^T A_{02})$$

Continue

$$\left(\begin{array}{c|c} A_{TL} & A_{TR} \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & a_{01} & A_{02} \\ \hline a_{10}^T & \alpha_{11} & a_{12}^T \\ \hline A_{20} & a_{21} & A_{22} \end{array}\right)$$

endwhile

Questions

- $\check{L}\check{U} = A + \Delta A$?
- $|\Delta A| \leq O(n)\mathbf{u}|\check{L}||\check{U}|$?
  $|\Delta A| \leq O(n)\mathbf{u}(|\check{L}||\check{U}| + |A|)$?

Invariants

- Loop-invariant:

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

- Error-invariant:

$$\left(\begin{array}{c|c} \check{L}_{TL}\check{U}_{TL}=A_{TL}+\Delta A_{TL} & \check{L}_{TL}\check{U}_{TR}=A_{TR}+\Delta A_{TR} \\ \hline \check{L}_{BL}\check{U}_{TL}=A_{BL}+\Delta A_{BL} & \end{array}\right)$$

Extended worksheet: Crout

$\check{L}_{00}\check{U}_{00} = A_{00} + \Delta A_{00}$	$\check{L}_{00}\check{u}_{01} = a_{01} + \delta a_{01}$	$\check{L}_{00}\check{U}_{02} = A_{02} + \Delta A_{02}$
$\check{l}_{10}^T\check{U}_{00} = a_{10}^T + \delta a_{10}^T$	--	--
$\check{L}_{20}\check{U}_{00} = A_{20} + \Delta A_{02}$	--	--

$$\upsilon_{11} := \alpha_{11} - l_{10}^T u_{01}$$
$$u_{12}^T := a_{12}^T - l_{10}^T U_{02}$$
$$l_{21} := (a_{21} - L_{20}u_{01})/\upsilon_{11}$$

Error Updates

$\check{L}_{00}\check{U}_{00} = A_{00} + \Delta A_{00}$	$\check{L}_{00}\check{u}_{01} = a_{01} + \delta a_{01}$	$\check{L}_{00}\check{U}_{02} = A_{02} + \Delta A_{02}$
$\check{l}_{10}^T\check{U}_{00} = a_{10}^T + \delta a_{10}^T$	$\check{l}_{10}^T\check{u}_{01} + \check{\upsilon}_{11} = \alpha_{11} + \delta\alpha_{11}$	$\check{l}_{10}^T\check{U}_{02} + \check{u}_{12}^T = a_{12}^T + \delta a_{12}^T$
$\check{L}_{20}\check{U}_{00} = A_{20} + \Delta A_{20}$	$\check{L}_{20}\check{u}_{01} + \check{l}_{21}\check{\upsilon}_{11} = a_{21} + \delta a_{21}$	--

Sub-goals

- $\upsilon_{11} := \alpha_{11} - l_{10}^T u_{01} \overset{?}{\Rightarrow} \check{l}_{10}^T\check{u}_{01} + \check{\upsilon}_{11} = \alpha_{11} + \delta\alpha_{11}$
- $u_{12}^T := a_{12}^T - l_{10}^T U_{02} \overset{?}{\Rightarrow} \check{l}_{10}^T\check{U}_{02} + \check{u}_{12}^T = a_{12}^T + \delta a_{12}^T$
- $l_{21} := (a_{21} - L_{20}u_{01})/\upsilon_{11} \overset{?}{\Rightarrow} \check{L}_{20}\check{u}_{01} + \check{l}_{21}\check{\upsilon}_{11} = a_{21} + \delta a_{21}$

- $\upsilon_{11} := \alpha_{11} - l_{10}^T u_{01}$

$$l_{10}^T u_{01} + \check{\upsilon}_{11} = \alpha_{11} + \delta\upsilon_{11}, \qquad |\delta\upsilon_{11}| \leq \gamma_{k+1} \left(\left| l_{10}^T \right| |u_{01}| + |\check{\upsilon}_{11}| \right)$$

- $u_{12}^T := a_{12}^T - l_{10}^T U_{02}$

$$l_{10}^T U_{02} + \check{u}_{12}^T = a_{12}^T + \delta u_{12}^T, \qquad |\delta u_{12}^T| \leq \gamma_{k+1} \left(\left| l_{10}^T \right| |U_{02}| + \left| \check{u}_{12}^T \right| \right)$$

- $l_{21} := (a_{21} - L_{20} u_{01}) / \upsilon_{11}$

$$L_{20} u_{01} + \upsilon_{11} \check{l}_{21} = a_{21} + \delta l_{21}, \qquad |\delta l_{21}| \leq \gamma_{k+1} \left(|L_{20}| \, |u_{01}| + \left| \check{l}_{21} \right| |\upsilon_{11}| \right)$$

$$\Rightarrow \quad \left(\begin{array}{c|c} \check{L}_{TL} \check{U}_{TL} = A_{TL} + \Delta A_{TL} & \check{L}_{TL} \check{U}_{TR} = A_{TR} + \Delta A_{TR} \\ \hline \check{L}_{BL} \check{U}_{TL} = A_{BL} + \Delta A_{BL} & \end{array} \right)$$

$$\Rightarrow \quad \check{L}\check{U} = A + \Delta A, \qquad |\Delta A| \leq \gamma_n |\check{L}||\check{U}|$$

Conclusions: blocked LU (sketch)

- Target operation: $LU = A$
- Algorithm: LU, TRSM$(\times 2)$, GEMM;
- Loop-invariant:

$$\left(\begin{array}{c|c} L_{TL}U_{TL} = A_{TL} & L_{TL}U_{TR} = A_{TR} \\ \hline L_{BL}U_{TL} = A_{BL} & A^{i}_{BR} = A_{BR} - L_{BL}U_{TR} \end{array}\right)$$

Conclusions: blocked LU (sketch)

- Target operation: $LU = A$
- Algorithm: $\texttt{LU}$, $\texttt{TRSM}(\times 2)$, $\texttt{GEMM}$;
- Loop-invariant:

$$\left(\begin{array}{c|c} L_{TL}U_{TL} = A_{TL} & L_{TL}U_{TR} = A_{TR} \\ \hline L_{BL}U_{TL} = A_{BL} & A^i_{BR} = A_{BR} - L_{BL}U_{TR} \end{array}\right)$$

Goal oriented

- Target analysis: $\check{L}\check{U} = A + \Delta A$
- Target bounds: $|\Delta A| \le \gamma_n |\check{L}||\check{U}|, \quad |\Delta A| \le \gamma_{\frac{n}{b}+b}(|\check{L}||\check{U}| + |A|)$
- Error-invariant:

$$\left(\begin{array}{c|c} \check{L}_{TL}\check{U}_{TL} = A_{TL} + \Delta A_{TL} & \check{L}_{TL}\check{U}_{TR} = A_{TR} + \Delta A_{TR} \\ \hline \check{L}_{BL}\check{U}_{TL} = A_{BL} + \Delta A_{BL} & \check{A}^i_{BR} = A_{BR} - \check{L}_{BL}\check{U}_{TR} + \Delta A^i_{BR} \end{array}\right) \wedge$$

$$\left|\left(\begin{array}{c|c} \Delta A_{TL} & \Delta A_{TR} \\ \hline \Delta A_{BL} & \Delta A^i_{BR} \end{array}\right)\right| \le \gamma_{\frac{k}{b}+b} \left(\begin{array}{c|c} |A_{TL}| + |\check{L}_{TL}||\check{U}_{TL}| & |A_{TR}| + |\check{L}_{TL}||\check{U}_{TR}| \\ \hline |A_{TL}| + |\check{L}_{BL}||\check{U}_{TL}| & |A_{BR}| + |\check{L}_{BL}||\check{U}_{TR}| \end{array}\right) \wedge$$

$size(A_{TL}) = k \times k$

Conclusions: blocked LU (sketch)

Modular

Composition of analyses

Subgoals:

- Recursive $\mathtt{LU}_{11} \stackrel{?}{\Rightarrow} \text{target}_{11}$
- $\mathtt{TRSM}_{12} \stackrel{?}{\Rightarrow} \text{target}_{12}$
- $\mathtt{TRSM}_{21} \stackrel{?}{\Rightarrow} \text{target}_{21}$
- $\mathtt{GEMM}_{22} \stackrel{?}{\Rightarrow} \text{target}_{22}$

Systematic

Prescribed sequence of steps

Sylvester? Undergraduate project

Acknowledgments

- Robert van de Geijn
- The FLAME team
- “Goal-oriented and modular stability analysis” (SIMAX)
  Dedicated to Pete on the occasion of his 70th bday

Financial support from the Deutsche Forschungsgemeinschaft (German Research Association) through grant GSC 111 is gratefully acknowledged.