Symmetric Indefinite Triangular Factorization Revealing the Rank Profile Matrix

Jean-Guillaume Dumas, Clément Pernet

Université Grenoble Alpes, Laboratoire Jean Kuntzmann, UMR CNRS

ISSAC'18, New York, USA
July 17, 2017

Supported by OpenDreamKit Horizon 2020 European RI project (#676541)

Context

Applications of symmetric Gaussian elimination

- Symmetric linear system solving
- Signature
- LLL: R factor of rational QR [Villard'12])

Context

Applications of symmetric Gaussian elimination

- Symmetric linear system solving
- Signature
- LLL: R factor of rational QR [Villard'12])

Compared to unsymmetric Gaussian elimination

- Save a factor of 2 in time complexity
- Invariants specific to symmetric matrices (signature)

Context

Applications of symmetric Gaussian elimination

- Symmetric linear system solving
- Signature
- LLL: R factor of rational QR [Villard'12])

Compared to unsymmetric Gaussian elimination

- Save a factor of 2 in time complexity
- Invariants specific to symmetric matrices (signature)

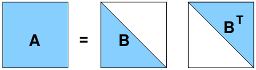
Motivation here

- `fsytrf`: finite field dense symmetric elimination in `fflas-ffpack`
- to be lifted for `LinBox` signature over $\mathbb{Z}$
- reduction to matrix product: $O(n^2 r^{\omega-2})$ and BLAS3
- investigate symmetric rank profile matrix and related pivoting

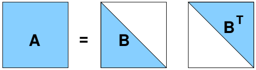
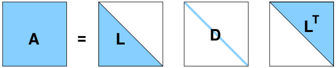
Outline

1. State of the art on symmetric factorizations
2. Rank profile and pivoting
3. Algorithms
4. The characteristic 2 case
5. Performance

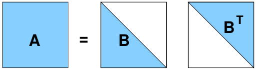
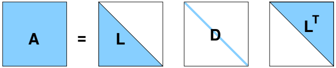
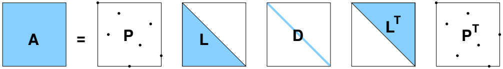
Symmetric factorizations

Symmetric Decomposition	Exists for
	Field with sqrt & Generic rank profile

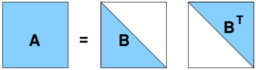
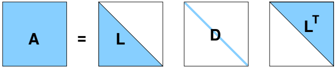
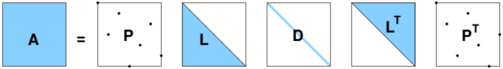
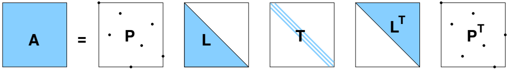
Symmetric factorizations

Symmetric Decomposition	Exists for
	Field with sqrt & Generic rank profile
	Generic rank profile

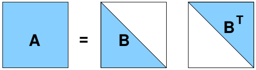
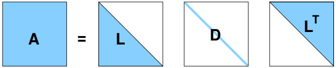
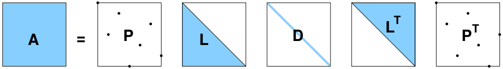
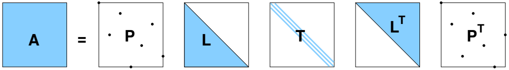
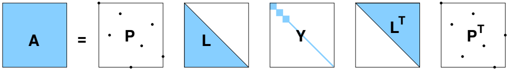
Symmetric factorizations

Symmetric Decomposition	Exists for
	Field with sqrt & Generic rank profile
	Generic rank profile
	No $\left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$-like blocks

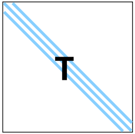
Symmetric factorizations

Symmetric Decomposition	Exists for
	Field with sqrt & Generic rank profile
	Generic rank profile
	No $\begin{bmatrix}0&1\\1&0\end{bmatrix}$-like blocks
	Any [Parlett-Reid 1970] T tridiagonal

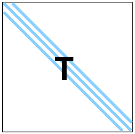
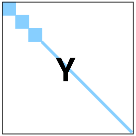
Symmetric factorizations

Symmetric Decomposition	Exists for
	Field with sqrt & Generic rank profile
	Generic rank profile
	No $\left[\begin{smallmatrix}0 & 1\\1 & 0\end{smallmatrix}\right]$-like blocks
	Any [Parlett-Reid 1970] T tridiagonal
	⋮
	Any [Bunch-Kaufmann 1977] Y with 1×1 and 2×2 blocks

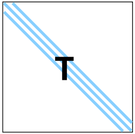
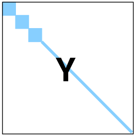
State of the art

Form	Properties		
	[Parlett-Reid 1970] Diagonal Pivoting Iterative $\frac{2}{3}n^3$	[Bunch-Parlett 1971] Full pivoting Iterative $\frac{2}{3}n^3$	[Aasen 1971] Partial pivoting Iterative $\frac{1}{3}n^3$

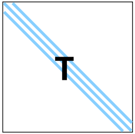
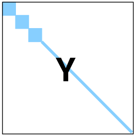
State of the art

Form	Properties		
	[Parlett-Reid 1970] Diagonal Pivoting Iterative $\frac{2}{3}n^3$	[Bunch-Parlett 1971] Full pivoting Iterative $\frac{2}{3}n^3$	[Aasen 1971] Partial pivoting Iterative $\frac{1}{3}n^3$
	[Bunch-Kaufmann 1977] Partial pivoting Iterative $\frac{1}{3}n^3$		

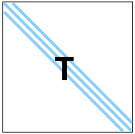
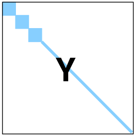
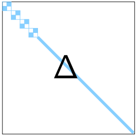
State of the art

Form	Properties		
	[Parlett-Reid 1970] Diagonal Pivoting Iterative $\frac{2}{3}n^3$	[Bunch-Parlett 1971] Full pivoting Iterative $\frac{2}{3}n^3$	[Aasen 1971] Partial pivoting Iterative $\frac{1}{3}n^3$
	[Bunch-Kaufmann 1977] Partial pivoting Iterative $\frac{1}{3}n^3$	[Shklarski-Toledo 2011] Partial pivoting Recursive (GRP hyp.) $\frac{1}{3}n^3$	

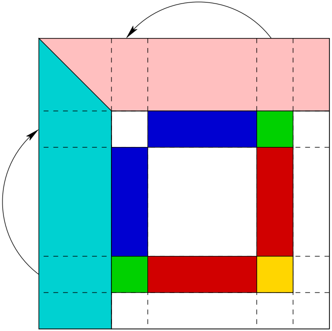
State of the art

Form	Properties		
	[Parlett-Reid 1970] Diagonal Pivoting Iterative $\frac{2}{3}n^3$	[Bunch-Parlett 1971] Full pivoting Iterative $\frac{2}{3}n^3$	[Aasen 1971] Partial pivoting Iterative $\frac{1}{3}n^3$
	[Bunch-Kaufmann 1977] Partial pivoting Iterative $\frac{1}{3}n^3$	[Shklarski-Toledo 2011] Partial pivoting Recursive (GRP hyp.) $\frac{1}{3}n^3$	[Yamazaki-Dongarra 2017] Partial pivoting Block Iterative $\frac{1}{3}n^3$

State of the art

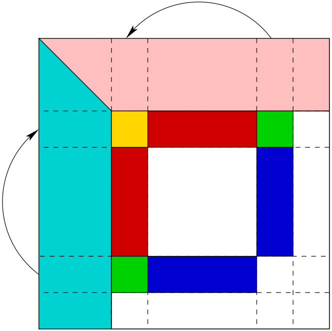
Form	Properties		
	[Parlett-Reid 1970] Diagonal Pivoting Iterative $\frac{2}{3}n^3$	[Bunch-Parlett 1971] Full pivoting Iterative $\frac{2}{3}n^3$	[Aasen 1971] Partial pivoting Iterative $\frac{1}{3}n^3$
	[Bunch-Kaufmann 1977] Partial pivoting Iterative $\frac{1}{3}n^3$	[Shklarski-Toledo 2011] Partial pivoting Recursive (GRP hyp.) $\frac{1}{3}n^3$	[Yamazaki-Dongarra 2017] Partial pivoting Block Iterative $\frac{1}{3}n^3$
	Here Pivoting revealing the rank profile matrix Recursive for any matrix $O(n^2 r^{\omega-2})$ (gives $\frac{1}{3}n^3$ when $rank{=}n$ & $\omega{=}3$)		

Symmetric pivoting



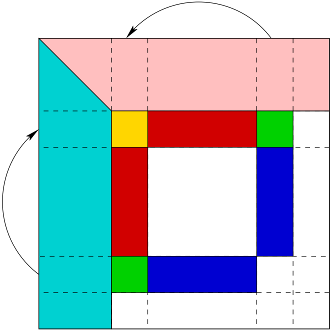
- Diagonal pivoting

Symmetric pivoting



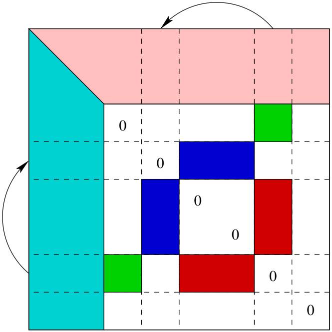
- Diagonal pivoting

Symmetric pivoting



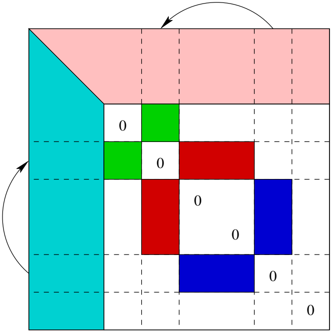
- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal

Symmetric pivoting



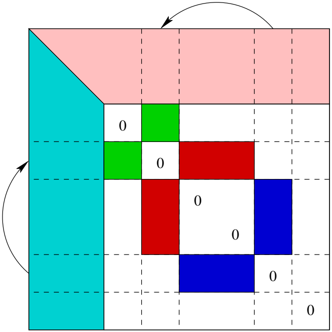
- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal

Symmetric pivoting



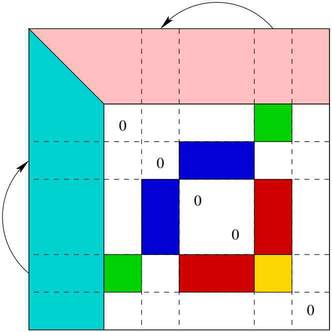
- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal

Symmetric pivoting



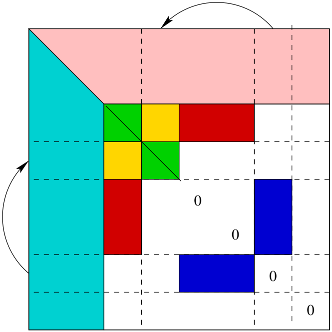
- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal
  $\Rightarrow L\Delta L^T$ with Δ block diagonal, 1×1 or 2×2 $\left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ blocks

Symmetric pivoting



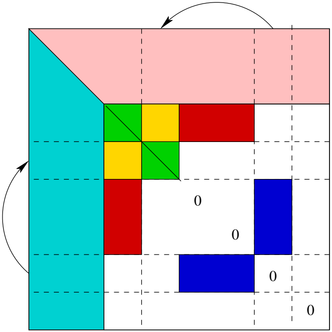
- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal
  $\Rightarrow L\Delta L^T$ with Δ block diagonal, 1×1 or 2×2 $\left[\begin{smallmatrix}0 & 1\\1 & 0\end{smallmatrix}\right]$ blocks
- Off-diagonal pivoting with non-zero diagonal

Symmetric pivoting



- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal
  $\Rightarrow L\Delta L^T$ with Δ block diagonal, 1×1 or 2×2 $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ blocks
- Off-diagonal pivoting with non-zero diagonal

Symmetric pivoting



- Diagonal pivoting
  $\Rightarrow LDL^T$ with D diagonal
- Off-diagonal pivoting with zero diagonal
  $\Rightarrow L\Delta L^T$ with Δ block diagonal, 1×1 or 2×2 $\left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ blocks
- Off-diagonal pivoting with non-zero diagonal
  $\Rightarrow LDL^T$ with D diagonal
  $\Rightarrow$ requires division by 2

The rank profile matrix

Rank Profiles

Given a matrix A of rank r:

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

The rank profile matrix

Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

The rank profile matrix

Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

The rank profile matrix

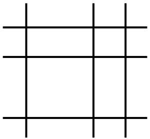
Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$



The rank profile matrix

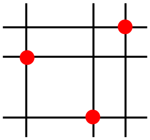
Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$



RPM (Rank Profile Matrix)

The unique $\mathcal{R}_A$ such that any pair of (i, j)-leading sub-matrix of $\mathcal{R}_A$ and of A have the same rank.

Pivoting revealing the rank profile matrix (unsymmetric)

[D., P., Sultan. ISSAC'15.] *Computing the Rank Profile Matrix.*

Definition

$A = PLUQ$ reveals the rank profile matrix if $P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q = \mathcal{R}_A$

Search	Row perm.	Col. perm.	RowRP	ColRP	$\mathcal{R}_A$	Instance
Row order	Transposition	Transposition	★			[IMH82] [JPS13]
Col. order	Transposition	Transposition		★		[KG85] [JPS13]
Lexico.	Transposition	Transposition	★			[Sto00]
Lexico.	Transposition	Rotation	★	★	★	[DPS15]
Lexico.	Rotation	Rotation	★	★	★	[DPS15]
Rev. lex.	Transposition	Transposition		★		[Sto00]
Rev. lex.	Rotation	Transposition	★	★	★	[DPS15]
Rev. lex.	Rotation	Rotation	★	★	★	[DPS15]
Product	Rotation	Transposition	★			[DPS15]
Product	Transposition	Rotation		★		[DPS15]
Product	Rotation	Rotation	★	★	★	[DPS13]

Pivoting revealing the rank profile matrix (symmetric case)

Definition

$A = PL\Delta L^T P^T$ reveals the rank profile matrix $\mathcal{R}_A$ if

$$P\Delta P^T = \mathrm{Diag}(d_1, \ldots, d_n)\mathcal{R}_A$$

Pivoting revealing the rank profile matrix (symmetric case)

Definition

$A = PL\Delta L^T P^T$ reveals the rank profile matrix $\mathcal{R}_A$ if

$$P\Delta P^T = \mathrm{Diag}(d_1, \ldots, d_n)\mathcal{R}_A$$

Pivoting revealing the rank profile matrix

1. Find pivot with minimal coordinates w.r.t.
   - lexicographic, reverse lexicographic or product order
2. Move it using cyclic rotations

Pivoting revealing the rank profile matrix (symmetric case)

Definition

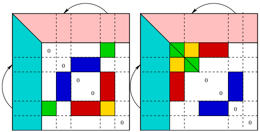
$A = PL\Delta L^T P^T$ reveals the rank profile matrix $\mathcal{R}_A$ if

$$P \Delta P^T = \text{Diag}(d_1, \ldots, d_n)\mathcal{R}_A$$

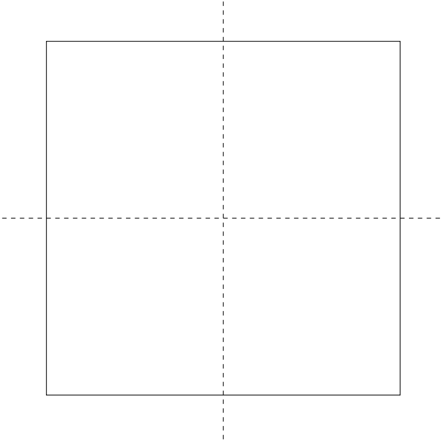
Pivoting revealing the rank profile matrix

1. Find pivot with minimal coordinates w.r.t.
   - lexicographic, reverse lexicographic or product order
2. Move it using cyclic rotations

⇒ Off-diagonal pivoting with non-zero diagonal
⇒ Requires division by 2.

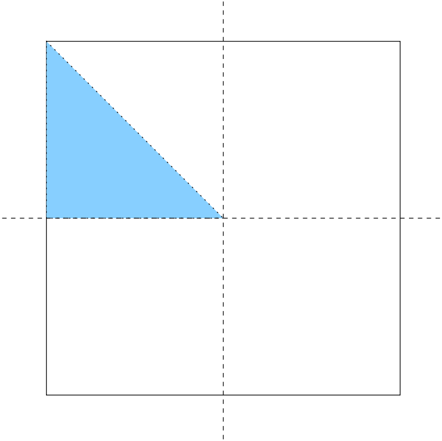


The recursive algorithm: full rank case



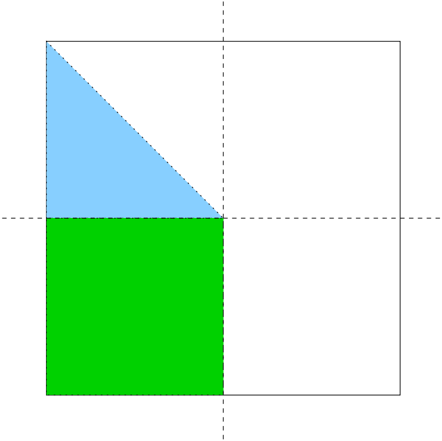
1. Split

The recursive algorithm: full rank case



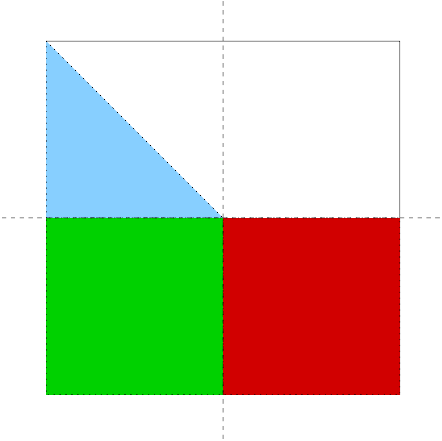
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$

The recursive algorithm: full rank case



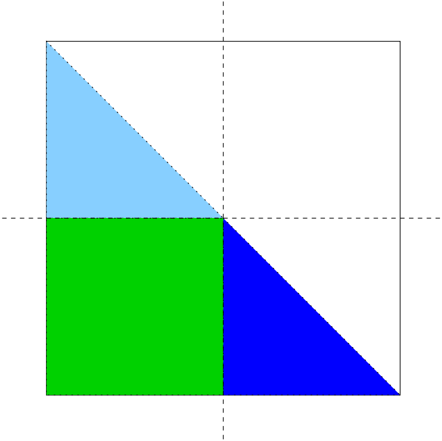
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$

The recursive algorithm: full rank case



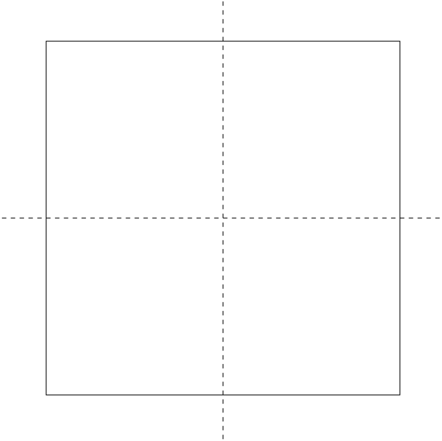
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. SYRK: $H \leftarrow A_{22} - GG^T$

The recursive algorithm: full rank case



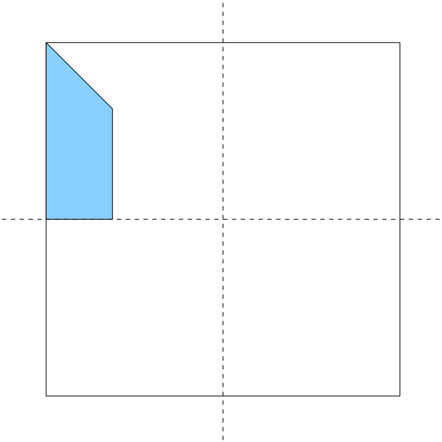
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. SYRK: $H \leftarrow A_{22} - GG^T$
5. Recursive call: $H = L_2 L_2^T$

The recursive algorithm: arbitrary rank case



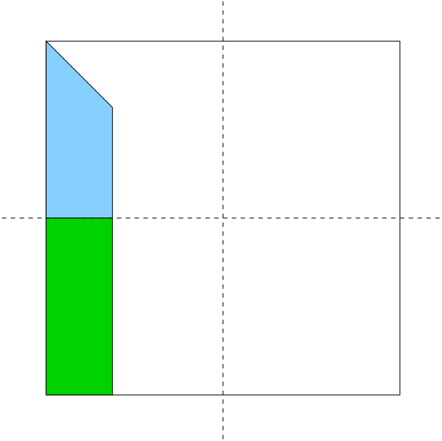
1. Split

The recursive algorithm: arbitrary rank case



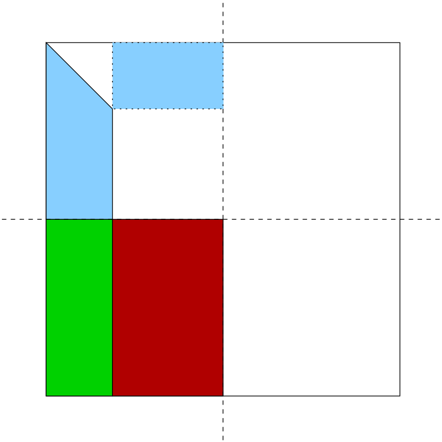
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$

The recursive algorithm: arbitrary rank case



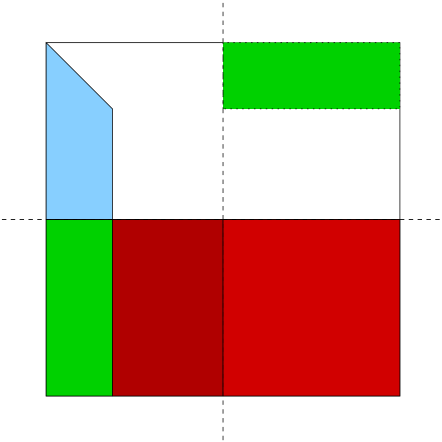
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$

The recursive algorithm: arbitrary rank case



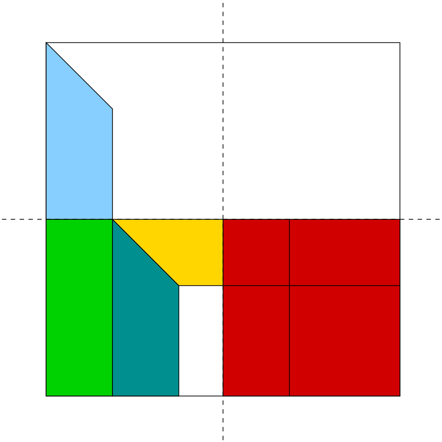
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$

The recursive algorithm: arbitrary rank case



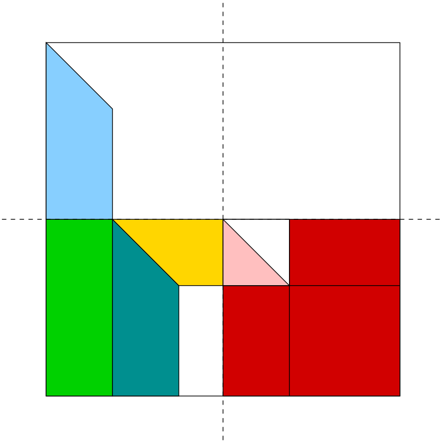
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$

The recursive algorithm: arbitrary rank case



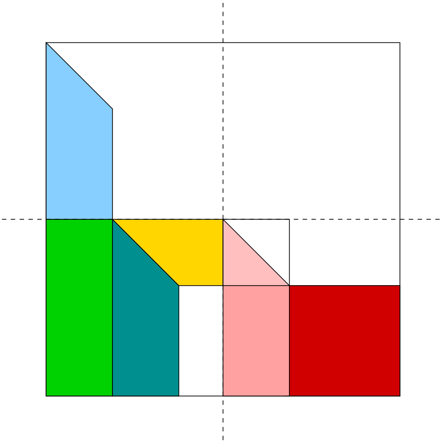
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$

The recursive algorithm: arbitrary rank case



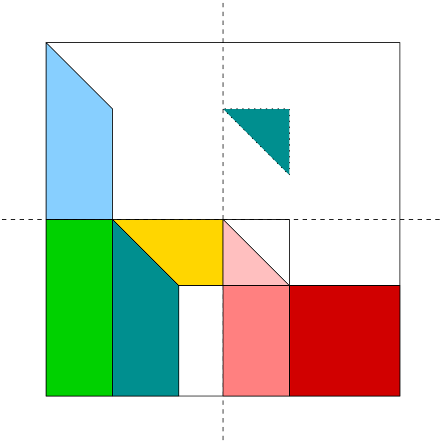
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$

The recursive algorithm: arbitrary rank case



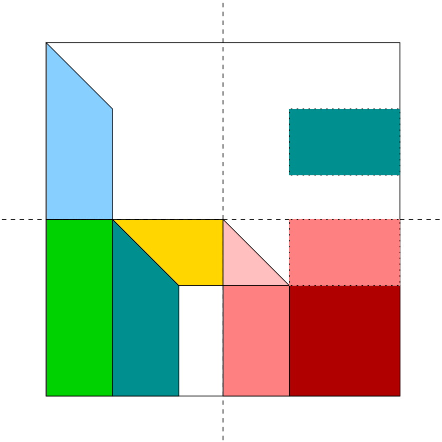
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$
8. TRMM: $H_{21} \leftarrow H_{21} - M_2 X$

The recursive algorithm: arbitrary rank case



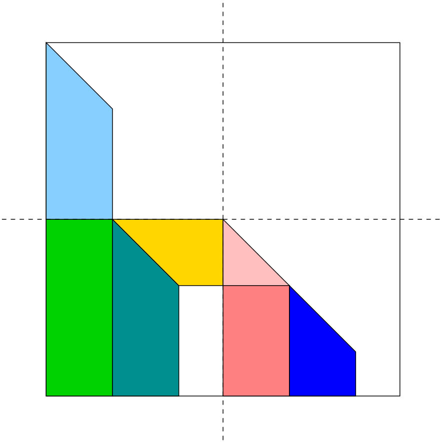
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$
8. TRMM: $H_{21} \leftarrow H_{21} - M_2 X$
9. TRSM: $H_{21} \leftarrow H_{21} L_2^{-T}$

The recursive algorithm: arbitrary rank case



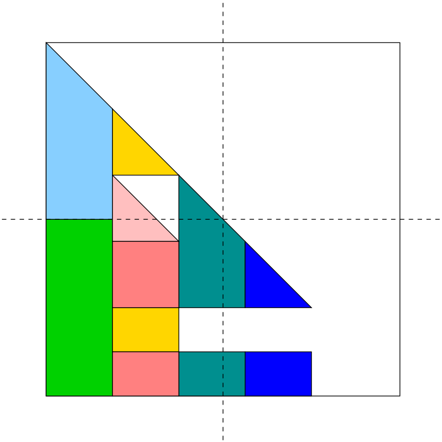
1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$
8. TRMM: $H_{21} \leftarrow H_{21} - M_2 X$
9. TRSM: $H_{21} \leftarrow H_{21} L_2^{-T}$
10. SYR2K: $H_{22} \leftarrow H_{22} - M_2 H_{21}^T - H_{21} M_2^T$

The recursive algorithm: arbitrary rank case



1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$
8. TRMM: $H_{21} \leftarrow H_{21} - M_2 X$
9. TRSM: $H_{21} \leftarrow H_{21} L_2^{-T}$
10. SYR2K: $H_{22} \leftarrow H_{22} - M_2 H_{21}^T - H_{21} M_2^T$
11. Recursive call: $H_{22} = L_3 L_3^T$

The recursive algorithm: arbitrary rank case



1. Split
2. Recursive call: $A_{11} = L_1 L_1^T$
3. TRSM: $G \leftarrow A_{21} L_1^{-1}$
4. GEMM: $K \leftarrow A_{22} - GM^T$
5. SYRK: $H \leftarrow A_{22} - GG^T$
6. PLUQ: $K = P_2 L_2 U_2 Q_2$
7. TRSYR2K: find X s.t. $L_2 X^T + X L_2^T = H_{11}$
8. TRMM: $H_{21} \leftarrow H_{21} - M_2 X$
9. TRSM: $H_{21} \leftarrow H_{21} L_2^{-T}$
10. SYR2K: $H_{22} \leftarrow H_{22} - M_2 H_{21}^T - H_{21} M_2^T$
11. Recursive call: $H_{22} = L_3 L_3^T$
12. Permutation

The `trsyr2k` routine

Problem

Given C symmetric and L lower triangular
Find X lower triangular such that

$$LX^T + XL^T = C$$

$$\begin{bmatrix} L_1 & \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} X_1^T & X_2^T \\ & X_3^T \end{bmatrix} + \begin{bmatrix} X_1 & \\ X_2 & X_3 \end{bmatrix} \begin{bmatrix} L_1^T & L_2^T \\ & L_3^T \end{bmatrix} = \begin{bmatrix} C_1 & C_2^T \\ C_2 & C_3 \end{bmatrix}.$$

The `trsyr2k` routine

Problem

Given C symmetric and L lower triangular
Find X lower triangular such that

$$LX^T + XL^T = C$$

$$\begin{bmatrix} L_1 & \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} X_1^T & X_2^T \\ & X_3^T \end{bmatrix} + \begin{bmatrix} X_1 & \\ X_2 & X_3 \end{bmatrix} \begin{bmatrix} L_1^T & L_2^T \\ & L_3^T \end{bmatrix} = \begin{bmatrix} C_1 & C_2^T \\ C_2 & C_3 \end{bmatrix}.$$

$$L_1 {X_1}^T + X_1 L_1^T = C_1$$

1. Find X_1 s.t. $L_1 X_1^T + X_1 L_1^T = C_1$ `recurse`

The `trsyr2k` routine

Problem

Given C symmetric and L lower triangular
Find X lower triangular such that

$$LX^T + XL^T = C$$

$$\begin{bmatrix} L_1 & \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} X_1^T & X_2^T \\ & X_3^T \end{bmatrix} + \begin{bmatrix} X_1 & \\ X_2 & X_3 \end{bmatrix} \begin{bmatrix} L_1^T & L_2^T \\ & L_3^T \end{bmatrix} = \begin{bmatrix} C_1 & C_2^T \\ C_2 & C_3 \end{bmatrix}.$$

$$L_2X_1^T + X_2L_1^T = C_2$$

1. Find X_1 s.t. $L_1X_1^T + X_1L_1^T = C_1$ — `recurse`
2. $Y \leftarrow C_2 - L_2X_1^T$ — `trmm`
3. $X_2 \leftarrow (L_1^T)^{-1}$ — `trsm`

The `trsyr2k` routine

Problem

Given C symmetric and L lower triangular
Find X lower triangular such that

$$LX^T + XL^T = C$$

$$\begin{bmatrix} L_1 & \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} X_1^T & X_2^T \\ & X_3^T \end{bmatrix} + \begin{bmatrix} X_1 & \\ X_2 & X_3 \end{bmatrix} \begin{bmatrix} L_1^T & L_2^T \\ & L_3^T \end{bmatrix} = \begin{bmatrix} C_1 & C_2^T \\ C_2 & C_3 \end{bmatrix}.$$

$$L_2X_2^T + X_2L_2^T + L_3X_3{}^T + X_3L_3^T = C_3$$

1. Find X_1 s.t. $L_1X_1^T + X_1L_1^T = C_1$ — `recurse`
2. $Y \leftarrow C_2 - L_2X_1^T$ — `trmm`
3. $X_2 \leftarrow (L_1^T)^{-1}$ — `trsm`
4. $Z \leftarrow C_3 - L_2X_2^T + X_2L_2$ — `syr2k`

The `trsyr2k` routine

Problem

Given C symmetric and L lower triangular
Find X lower triangular such that

$$LX^T + XL^T = C$$

$$\begin{bmatrix} L_1 & \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} X_1^T & X_2^T \\ & X_3^T \end{bmatrix} + \begin{bmatrix} X_1 & \\ X_2 & X_3 \end{bmatrix} \begin{bmatrix} L_1^T & L_2^T \\ & L_3^T \end{bmatrix} = \begin{bmatrix} C_1 & C_2^T \\ C_2 & C_3 \end{bmatrix}.$$

$$L_2X_2^T + X_2L_2^T + L_3{X_3}^T + X_3L_3^T = C_3$$

1. Find X_1 s.t. $L_1X_1^T + X_1L_1^T = C_1$ `recurse`
2. $Y \leftarrow C_2 - L_2X_1^T$ `trmm`
3. $X_2 \leftarrow (L_1^T)^{-1}$ `trsm`
4. $Z \leftarrow C_3 - L_2X_2^T + X_2L_2$ `syr2k`
5. Find X_3 s.t. $L_3X_3^T + X_3L_3^T = Z$ `recurse`

Problem in characteristic 2

Consider $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix}$ (with $\mathcal{R} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$)

Diagonal pivoting: $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix} = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{c}{d} & 1 \end{bmatrix} \begin{bmatrix} d & \\ & -\frac{c^2}{d} \end{bmatrix} \begin{bmatrix} 1 & \frac{c}{d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}$

Problem in characteristic 2

Consider $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix}$ (with $\mathcal{R} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$)

Diagonal pivoting: $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix} = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{c}{d} & 1 \end{bmatrix} \begin{bmatrix} d & \\ & -\frac{c^2}{d} \end{bmatrix} \begin{bmatrix} 1 & \frac{c}{d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}$

Off-diagonal pivoting: $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{d}{2c} & 1 \end{bmatrix} \begin{bmatrix} 0 & c \\ c & 0 \end{bmatrix} \begin{bmatrix} 1 & \frac{d}{2c} \\ 0 & 1 \end{bmatrix}$

Problem in characteristic 2

Consider $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix}$ (with $\mathcal{R} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$)

Diagonal pivoting: $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix} = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{c}{d} & 1 \end{bmatrix} \begin{bmatrix} d & \\ & -\frac{c^2}{d} \end{bmatrix} \begin{bmatrix} 1 & \frac{c}{d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}$

Off-diagonal pivoting: $\begin{bmatrix} 0 & c \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{d}{2c} & 1 \end{bmatrix} \begin{bmatrix} 0 & c \\ c & 0 \end{bmatrix} \begin{bmatrix} 1 & \frac{d}{2c} \\ 0 & 1 \end{bmatrix}$

Diagonal pivoting breaks RPM-revealing property
- $\Rightarrow$ off-diagonal pivoting required
- $\Rightarrow$ division by 2 required

Problem in characteristic 2

Consider $\begin{bmatrix}0 & c\\ c & d\end{bmatrix}$ (with $\mathcal{R} = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$)

Diagonal pivoting: $\begin{bmatrix}0 & c\\ c & d\end{bmatrix} = \begin{bmatrix} & 1\\ 1 & \end{bmatrix} \begin{bmatrix}1 & 0\\ \frac{c}{d} & 1\end{bmatrix} \begin{bmatrix}d & \\ & -\frac{c^2}{d}\end{bmatrix} \begin{bmatrix}1 & \frac{c}{d}\\ 0 & 1\end{bmatrix} \begin{bmatrix} & 1\\ 1 & \end{bmatrix}$

Off-diagonal pivoting: $\begin{bmatrix}0 & c\\ c & d\end{bmatrix} = \begin{bmatrix}1 & 0\\ \frac{d}{2c} & 1\end{bmatrix} \begin{bmatrix}0 & c\\ c & 0\end{bmatrix} \begin{bmatrix}1 & \frac{d}{2c}\\ 0 & 1\end{bmatrix}$

Diagonal pivoting breaks RPM-revealing property

- $\Rightarrow$ off-diagonal pivoting required
- $\Rightarrow$ division by 2 required

In characteristic 2

$\Rightarrow$ In general there is no RPM-revealing $P \cdot L \cdot \Delta \cdot L^T \cdot P^T$!

Characteristic 2: Preserve $\left[\begin{smallmatrix}0 & 1\\ 1 & 1\end{smallmatrix}\right]$ factors

- Compute instead $A = P \cdot L \cdot D \cdot \Psi \cdot L^T \cdot P^T$:

$\Rightarrow$ D diagonal, Ψ block diagonal with 1×1, $\left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ and $\left[\begin{smallmatrix}0 & 1\\ 1 & 1\end{smallmatrix}\right]$ blocks

Characteristic 2: Preserve $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ factors

- Compute instead $A = P \cdot L \cdot D \cdot \Psi \cdot L^T \cdot P^T$:

$\Rightarrow$ D diagonal, Ψ block diagonal with 1×1, $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ and $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ blocks

- From this intermediate form:
  - Either recover $\mathcal{R}_A = P \cdot \mathcal{R}_\Psi \cdot P^T$;
  - Or use:

$$\begin{bmatrix} & c \\ c & d \end{bmatrix} = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \cdot \begin{bmatrix} 1 & \\ c/d & 1 \end{bmatrix} \cdot \begin{bmatrix} d & \\ & -c^2/d \end{bmatrix} \cdot \begin{bmatrix} 1 & c/d \\ & 1 \end{bmatrix} \cdot \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}$$

☺ + commuting

$\Rightarrow$ $\tilde{P} \cdot \tilde{L} \cdot \Delta \cdot \tilde{L}^T \cdot \tilde{P}^T$ symmetric factorization (but $\mathcal{R}_A$ is lost)

Iterative base case

Practical performance:

⚠ Stop recursion (less mod reductions/data move on small matrices)

Iterative base case

Practical performance:

⚠ Stop recursion (less mod reductions/data move on small matrices)

⇒ iterative base case and cascading

- pivot search minimizing the lexicographic order
- cyclic shifts on the row and columns
- Crout elimination schedule

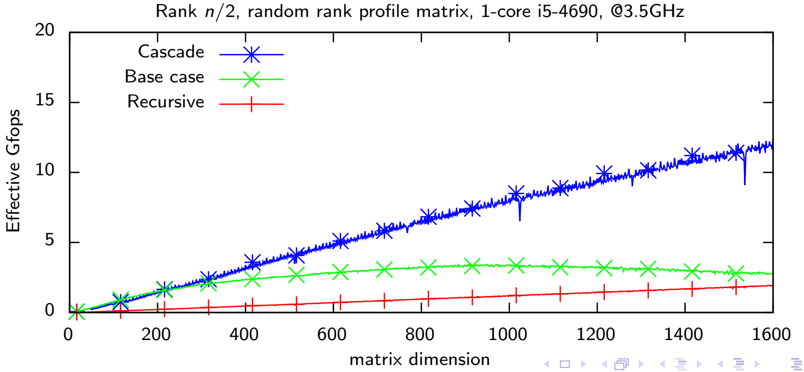
Iterative base case

Practical performance:

⚠ Stop recursion (less mod reductions/data move on small matrices)

⇒ iterative base case and cascading

- pivot search minimizing the lexicographic order
- cyclic shifts on the row and columns
- Crout elimination schedule



LAPACK vs FFPACK modulo 8 388 593

	LAPACK		FFPACK	
n	dgetrf (LU)	dsytrf (LDLT)	fgetrf (LU)	fsytrf (LDLT)
5000	2.01s	1.60s	3.90s	1.59s
10000	14.95s	11.98s	24.12s	10.90s

LAPACK vs FFPACK modulo 8 388 593

	LAPACK		FFPACK	
n	dgetrf (LU)	dsytrf (LDLT)	fgetrf (LU)	fsytrf (LDLT)
5000	2.01s	1.60s	3.90s	1.59s
10000	14.95s	11.98s	24.12s	10.90s