Improved generators for quasiseparable matrices

hal-01264131

Clément Pernet

Univ. Grenoble Alpes,
Laboratoire de l'Informatique du Parallélisme (Univ. de Lyon, Inria)

Structured Matrix Days,
Limoges, France
May 10, 2016

Introduction

Two main types of structured matrices

Rank displacement: Hankel, Toeplitz, Vandermonde, Cauchy, etc

- arising from polynomial operations (evaluation, interpolation, etc)
- Rank displacement: rank$(A - PAQ)$ for $P, Q \in \{\text{Diag}, \text{Cyclic shift}\}$
- Structure: rank$(A - PAQ)$ small $\rightsquigarrow A - PAQ = LR$

Rank structure: Tridiagonal, Hessenberg, semiseparable, quasiseparable

- Structure: small rank in the upper/lower triangular part
- no closed form expression

Quasiseparable matrices

$$
\begin{bmatrix}
4 & 2 & 3 & & & & & & & \\
4 & 0 & 1 & 4 & & & & & & \\
4 & 3 & 5 & 6 & 3 & & & & & \\
 & 5 & 1 & 1 & 4 & 5 & & & & \\
 & & 4 & 3 & 1 & 1 & 4 & & & \\
 & & & 4 & 0 & 5 & 1 & 2 & & \\
 & & & & 4 & 3 & 1 & 2 & 6 & \\
 & & & & & 2 & 1 & 3 & 1 & 1 \\
 & & & & & & 1 & 5 & 4 & 3 \\
 & & & & & & & 3 & 0 & 4
\end{bmatrix}^{-1}
=
\begin{bmatrix}
2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\
6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\
3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\
6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\
0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\
2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\
6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\
0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6
\end{bmatrix}
$$

Quasiseparable matrices

$$
\begin{bmatrix}
4 & 2 & 3 & & & & & & & \\
4 & 0 & 1 & 4 & & & & & & \\
4 & 3 & 5 & 6 & 3 & & & & & \\
 & 5 & 1 & 1 & 4 & 5 & & & & \\
 & & 4 & 3 & 1 & 1 & 4 & & & \\
 & & & 4 & 0 & 5 & 1 & 2 & & \\
 & & & & 4 & 3 & 1 & 2 & 6 & \\
 & & & & & 2 & 1 & 3 & 1 & 1 \\
 & & & & & & 1 & 5 & 4 & 3 \\
 & & & & & & & 3 & 0 & 4
\end{bmatrix}^{-1}
=
\begin{bmatrix}
2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\
6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\
3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\
6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\
0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\
2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\
6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\
0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6
\end{bmatrix}
$$

rank $= 1$ rank $= 1$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$
\begin{bmatrix}
4 & 2 & 3 & & & & & & & \\
4 & 0 & 1 & 4 & & & & & & \\
4 & 3 & 5 & 6 & 3 & & & & & \\
 & 5 & 1 & 1 & 4 & 5 & & & & \\
 & & 4 & 3 & 1 & 1 & 4 & & & \\
 & & & 4 & 0 & 5 & 1 & 2 & & \\
 & & & & 4 & 3 & 1 & 2 & 6 & \\
 & & & & & 2 & 1 & 3 & 1 & 1 \\
 & & & & & & 1 & 5 & 4 & 3 \\
 & & & & & & & 3 & 0 & 4
\end{bmatrix}^{-1}
=
\begin{bmatrix}
2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\
6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\
3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\
6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\
0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\
2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\
6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\
0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6
\end{bmatrix}
$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

$\text{rank} = 2$ $\qquad$ $\text{rank} = 2$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$
\begin{bmatrix}
4 & 2 & 3 & & & & & & & \\
4 & 0 & 1 & 4 & & & & & & \\
4 & 3 & 5 & 6 & 3 & & & & & \\
 & 5 & 1 & 1 & 4 & 5 & & & & \\
 & & 4 & 3 & 1 & 1 & 4 & & & \\
 & & & 4 & 0 & 5 & 1 & 2 & & \\
 & & & & 4 & 3 & 1 & 2 & 6 & \\
 & & & & & 2 & 1 & 3 & 1 & 1 \\
 & & & & & & 1 & 5 & 4 & 3 \\
 & & & & & & & 3 & 0 & 4
\end{bmatrix}^{-1}
=
\begin{bmatrix}
2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\
6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\
3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\
6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\
0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\
2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\
6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\
0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6
\end{bmatrix}
$$

rank $= 2$ rank $= 2$

Quasiseparable matrices

$$\begin{bmatrix}
4 & 2 & 3 & & & & & & & \\
4 & 0 & 1 & 4 & & & & & & \\
4 & 3 & 5 & 6 & 3 & & & & & \\
 & 5 & 1 & 1 & 4 & 5 & & & & \\
 & & 4 & 3 & 1 & 1 & 4 & & & \\
 & & & 4 & 0 & 5 & 1 & 2 & & \\
 & & & & 4 & 3 & 1 & 2 & 6 & \\
 & & & & & 2 & 1 & 3 & 1 & 1 \\
 & & & & & & 1 & 5 & 4 & 3 \\
 & & & & & & & 3 & 0 & 4
\end{bmatrix}^{-1}
=
\begin{bmatrix}
2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\
6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\
3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\
6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\
0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\
2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\
6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\
0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\
1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6
\end{bmatrix}$$

$\text{rank} = 1$ $\qquad$ $\text{rank} = 1$

Quasiseparable matrices

$$\begin{bmatrix} 4 & 2 & 3 & & & & & & & \\ 4 & 0 & 1 & 4 & & & & & & \\ 4 & 3 & 5 & 6 & 3 & & & & & \\ & 5 & 1 & 1 & 4 & 5 & & & & \\ & & 4 & 3 & 1 & 1 & 4 & & & \\ & & & 4 & 0 & 5 & 1 & 2 & & \\ & & & & 4 & 3 & 1 & 2 & 6 & \\ & & & & & 2 & 1 & 3 & 1 & 1 \\ & & & & & & 1 & 5 & 4 & 3 \\ & & & & & & & 3 & 0 & 4 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

rank $= 1$ rank $= 1$

Definition

M $\in \mathrm{K}^{n\times n}$ is (r_L, r_U)-quasiseparable if $\begin{cases} \text{rank}(\mathsf{M}_{k+1..n,1..k}) & \leq & r_L \\ \text{rank}(\mathsf{M}_{1..k,k+1..n}) & \leq & r_U \end{cases}$

Applications

Numerical linear algebra: naturally occur in solving

- generalized eigenvalue problems,
- PDE's, etc

Applications

Numerical linear algebra: naturally occur in solving

- generalized eigenvalue problems,
- PDE's, etc

Computer algebra: over $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{Z}/\mathrm{p}\mathbb{Z}$

- `LinBox`: black-box exact linear algebra library,
- shows up in algorithms for the Frobenius normal form, characteristic polynomial over $\mathbb{Z}/p\mathbb{Z}$

Applications

Numerical linear algebra: naturally occur in solving

- generalized eigenvalue problems,
- PDE's, etc

Computer algebra: over $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{Z}/\mathrm{p}\mathbb{Z}$

- `LinBox`: black-box exact linear algebra library,
- shows up in algorithms for the Frobenius normal form, characteristic polynomial over $\mathbb{Z}/p\mathbb{Z}$

Complexity

Displacement rank structure:

- Formerly $\tilde{O}(r^2 n)$ time and space [Pan01]
- with fast matrix arithmetic: $\tilde{O}(r^{\omega-1} n)$ time [BJS08]

Applications

Numerical linear algebra: naturally occur in solving

- generalized eigenvalue problems,
- PDE's, etc

Computer algebra: over $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{Z}/\text{p}\mathbb{Z}$

- `LinBox`: black-box exact linear algebra library,
- shows up in algorithms for the Frobenius normal form, characteristic polynomial over $\mathbb{Z}/p\mathbb{Z}$

Complexity

Displacement rank structure:

- Formerly $\tilde{O}(r^2 n)$ time and space [Pan01]
- with fast matrix arithmetic: $\tilde{O}(r^{\omega-1} n)$ time [BJS08]

Quasiseparable structure

- $O(r^2 n)$ time and space [EG99]
- Seeking to reduce the exponent in r

Structured representation of a quasiseparable matrix

[EG99]

$$\mathsf{M}_{i,j} = \begin{cases} p(i)^T \mathsf{a}(i-1)\ldots\mathsf{a}(j+1)q(j), & 1 \leq j < i \leq n \\ d(i), & 1 \leq i = j \leq n \\ g(i)^T \mathsf{b}(i+1)\ldots\mathsf{b}(i-1)h(j), & 1 \leq i < j \leq n \end{cases}$$

where for $i \in \{1..n-1\}$

- $p(i), q(i) \in \mathrm{K}^{r_L}$, $g(i), h(i) \in \mathrm{K}^{r_U}$,
- $\mathsf{a}(i) \in \mathrm{K}^{r_L \times r_L}$, $\mathsf{b}(i) \in \mathrm{K}^{r_U \times r_U}$,

Structured representation of a quasiseparable matrix

[EG99]

$$\mathsf{M}_{i,j} = \begin{cases} p(i)^T \mathsf{a}(i-1)\dots\mathsf{a}(j+1)q(j), & 1 \le j < i \le n \\ d(i), & 1 \le i = j \le n \\ g(i)^T \mathsf{b}(i+1)\dots\mathsf{b}(i-1)h(j), & 1 \le i < j \le n \end{cases}$$

where for $i \in \{1..n-1\}$

- $p(i), q(i) \in \mathrm{K}^{r_L}$, $g(i), h(i) \in \mathrm{K}^{r_U}$,
- $\mathsf{a}(i) \in \mathrm{K}^{r_L \times r_L}$, $\mathsf{b}(i) \in \mathrm{K}^{r_U \times r_U}$,

Size	$O(nr^2)$
Computing the generator	$O(n^2)$

Structured representation of a quasiseparable matrix

[EG99]

$$\mathsf{M}_{i,j} = \begin{cases} p(i)^T \mathsf{a}(i-1)\ldots\mathsf{a}(j+1)q(j), & 1 \le j < i \le n \\ d(i), & 1 \le i = j \le n \\ g(i)^T \mathsf{b}(i+1)\ldots\mathsf{b}(i-1)h(j), & 1 \le i < j \le n \end{cases}$$

where for $i \in \{1..n-1\}$

- $p(i), q(i) \in \mathrm{K}^{r_L}$, $g(i), h(i) \in \mathrm{K}^{r_U}$,
- $\mathsf{a}(i) \in \mathrm{K}^{r_L \times r_L}$, $\mathsf{b}(i) \in \mathrm{K}^{r_U \times r_U}$,

Size	$O(nr^2)$
Computing the generator	$O(n^2)$
Vector apply	$O(r^2 n)$ field ops

Structured representation of a quasiseparable matrix

[EG99]

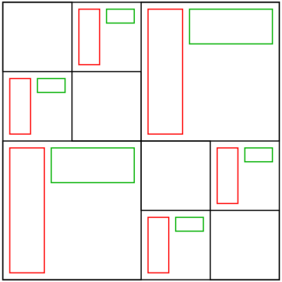
$$\mathsf{M}_{i,j} = \begin{cases} p(i)^T \mathsf{a}(i-1)\ldots\mathsf{a}(j+1)q(j), & 1 \le j < i \le n \\ d(i), & 1 \le i = j \le n \\ g(i)^T \mathsf{b}(i+1)\ldots\mathsf{b}(i-1)h(j), & 1 \le i < j \le n \end{cases}$$

where for $i \in \{1..n-1\}$

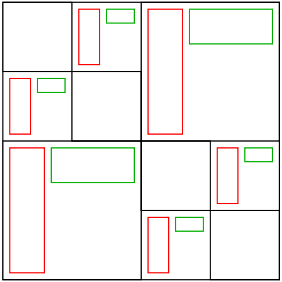
- $p(i), q(i) \in \mathrm{K}^{r_L}$, $\quad g(i), h(i) \in \mathrm{K}^{r_U}$,
- $\mathsf{a}(i) \in \mathrm{K}^{r_L \times r_L}$, $\quad \mathsf{b}(i) \in \mathrm{K}^{r_U \times r_U}$,

Size	$O(nr^2)$
Computing the generator	$O(n^2)$
Vector apply	$O(r^2 n)$ field ops
Quasisep $\times$ Quasisep	$O(r^3 n)$ field ops

Hierarchically Semi-Separable representation

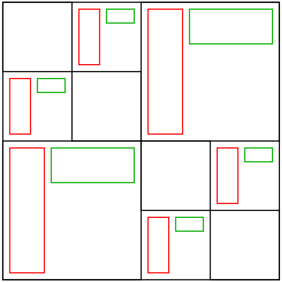


Hierarchically Semi-Separable representation



	[CGP06]
Size	$O(n^2)$ (peak)
Computing the generator Vector apply	$O(n^2)$?

Hierarchically Semi-Separable representation



	[CGP06]	[LHC16]
Size	$O(n^2)$ (peak)	$O(rn \log n)$
Computing the generator	$O(n^2)$	$O(rn \log n)$
Vector apply	?	$O(rn \log n)$ field ops

Contribution

Connecting notions of **quasiseparability** and **rank profile matrix** [DPS15]

Contribution

Connecting notions of **quasiseparability** and **rank profile matrix** [DPS15]

1. Compute the quasiseparable orders in $O(r^{\omega-2}n^2)$
2. Two alternative structured representations
   1. binary tree of PLUQ: $O(rn\log\frac{n}{r})$ space
   2. Bruhat generator: $O(rn)$ space
3. algorithms to compute these representations,
4. algorithms for Quasisep $\times$ vector and Quasisep $\times$ Quasisep

Contribution

Connecting notions of **quasiseparability** and **rank profile matrix** [DPS15]

1. Compute the quasiseparable orders in $O(r^{\omega-2}n^2)$
2. Two alternative structured representations
   1. binary tree of PLUQ: $O(rn\log\frac{n}{r})$ space
   2. Bruhat generator: $O(rn)$ space
3. algorithms to compute these representations,
4. algorithms for Quasisep $\times$ vector and Quasisep $\times$ Quasisep

	[EG99]	[Per16]
Size	$O(r^2n)$	$O(rn)$
Computing the generator	$O(n^2)$	$O(r^{\omega-2}n^2)$
Vector apply	$O(r^2n)$	$O(rn)$

Contribution

Connecting notions of **quasiseparability** and **rank profile matrix** [DPS15]

1. Compute the quasiseparable orders in $O(r^{\omega-2}n^2)$
2. Two alternative structured representations
   1. binary tree of PLUQ: $O(rn\log\frac{n}{r})$ space
   2. Bruhat generator: $O(rn)$ space
3. algorithms to compute these representations,
4. algorithms for Quasisep $\times$ vector and Quasisep $\times$ Quasisep

	[EG99]	[Per16]
Size	$O(r^2n)$	$O(rn)$
Computing the generator	$O(n^2)$	$\tilde{O}(n^2+r^\omega)$Prob ?
Vector apply	$O(r^2n)$	$O(rn)$

Contribution

Connecting notions of **quasiseparability** and **rank profile matrix** [DPS15]

1. Compute the quasiseparable orders in $O(r^{\omega-2}n^2)$
2. Two alternative structured representations
   1. binary tree of PLUQ: $O(rn\log\frac{n}{r})$ space
   2. Bruhat generator: $O(rn)$ space
3. algorithms to compute these representations,
4. algorithms for Quasisep $\times$ vector and Quasisep $\times$ Quasisep

	[EG99]	[Per16]
Size	$O(r^2n)$	$O(rn)$
Computing the generator	$O(n^2)$	$\tilde{O}(n^2+r^\omega)$Prob ?
Vector apply	$O(r^2n)$	$O(rn)$
Quasisep $\times$ Quasisep	$O(r^3n)$	$O(r^{\omega-2}n^2)$

Outline

1. The rank profile matrix
   - Definition
   - Computing the rank profile matrix
2. Computing the orders of quasiseparability
3. More efficient generators

Rank profiles

Definition (Row Rank Profile: RowRP)

Given $A \in \mathsf{K}^{m\times n}, r = \mathsf{rank}(A)$.

informally: *first* r linearly independent rows

formally: lexico-minimal list of r indices of linearly independant rows.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank profiles

Definition (Row Rank Profile: RowRP)

Given $A \in \mathsf{K}^{m\times n}, r = \mathsf{rank}(A)$.

informally: *first* r linearly independent rows

formally: lexico-minimal list of r indices of linearly independant rows.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 3
RowRP = {1,2,4}

Rank profiles

Definition (Column Rank Profile: ColRP)

Given $A \in \mathsf{K}^{m\times n}, r = \mathsf{rank}(A)$.

informally: *first* r linearly independent columns

formally: lexico-minimal list of r linearly independant columns.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 3
RowRP = {1,2,4}
ColRP = {1,2,3}

Rank profiles

Definition (Column Rank Profile: ColRP)

Given $A \in \mathsf{K}^{m\times n}, r = \mathsf{rank}(A)$.

informally: *first* r linearly independent columns

formally: lexico-minimal list of r linearly independant columns.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 3
RowRP = {1,2,4}
ColRP = {1,2,3} → Generic ColRP.

Rank profiles

Definition (Column Rank Profile: ColRP)

Given $A \in \mathsf{K}^{m\times n}, r = \text{rank}(A)$.

informally: *first* r linearly independent columns

formally: lexico-minimal list of r linearly independant columns.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 3
RowRP = {1,2,4}
ColRP = {1,2,3} → Generic ColRP.

Generic Rank Profile: first r leading principal minors $\neq 0$

Generic rank profile $\not\Leftarrow\Rightarrow$ Generic Row RP and Genric ColRP

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

RowRP = ColRP = {1,2}

Rank profiles

Definition (Column Rank Profile: ColRP)

Given $A \in \mathsf{K}^{m\times n}, r = \text{rank}(A)$.

informally: *first* r linearly independent columns

formally: lexico-minimal list of r linearly independant columns.

Example

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 3
RowRP = {1,2,4}
ColRP = {1,2,3} → Generic ColRP.

Generic Rank Profile: first r leading principal minors $\neq 0$

Generic rank profile $\not\Leftarrow\Rightarrow$ Generic Row RP and Genric ColRP

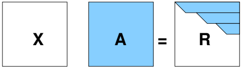
$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

RowRP = ColRP = {1,2}
But $|A_{1,1}| = 0$

Relation to echelon forms

Transformation to echelon form

$\forall A \; \exists X$ non-singular s.t.



Relation to echelon forms:

- ColRP unchanged by left multiplication with an invertible matrix

ColRP $=$ pivot columns in the **row** echelon form

Computing rank profiles

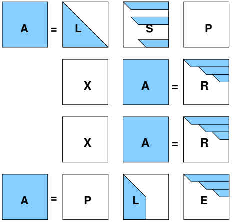
Via Gaussian elimination revealing row echelon forms:

[Ibarra, Moran and Hui 82]

[Keller-Gehrig 85]

[Storjohann 00]

[Jeannerod, P. and Storjohann 13]



Computing rank profiles

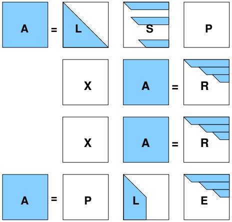
Via Gaussian elimination revealing row echelon forms:

[Ibarra, Moran and Hui 82]

[Keller-Gehrig 85]

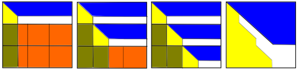
[Storjohann 00]

[Jeannerod, P. and Storjohann 13]



Lessons learned (or what we thought was necessary):

- treat rows in order
- exhaust all columns before considering the next row
- **slab** block splitting (recursive or iterative)



⇝ similar to partial pivoting

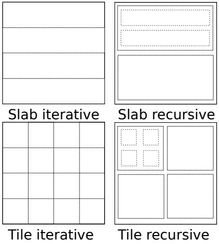
Motivation

Need more flexible blocking

Slab blocking

- leads to inefficient memory access patterns
- is harder to parallelize

Tile blocking instead ?



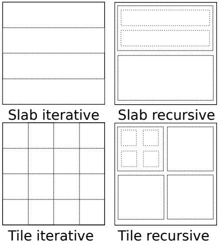
Motivation

Need more flexible blocking

Slab blocking

- leads to inefficient memory access patterns
- is harder to parallelize

Tile blocking instead ?



Gathering linear independence invariants

Two ways to look at a matrix (looking left or right):

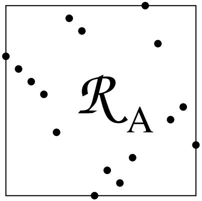
- Row rank profile, column echelon form
- Column rank profile, row echelon form

Unique invariant?

The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m\times n}$.
There exists a unique, $m \times n$, rank(A)-sub-permutation matrix $\mathcal{R}^A$ of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of A.



The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m \times n}$.
There exists a unique, $m \times n$, rank(A)-sub-permutation matrix $\mathcal{R}^A$ of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of A.

Example

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m \times n}$.
There exists a unique, $m \times n$, rank(A)-sub-permutation matrix $\mathcal{R}^A$ of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of A.

Example

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m \times n}$.
There exists a unique, $m \times n$, *rank*(A)*-sub-permutation matrix* $\mathcal{R}^A$ *of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of* A.

Example

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m \times n}$.
There exists a unique, $m \times n$, rank(A)-sub-permutation matrix $\mathcal{R}^A$ of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of A.

Example

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The rank profile Matrix

Theorem

Let $A \in \mathrm{F}^{m\times n}$.
There exists a unique, $m \times n$, rank(A)-sub-permutation matrix $\mathcal{R}^A$ of which every leading sub-matrix has the same rank as the corresponding leading sub-matrix of A.

Example

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Properties of the rank profile matrix

Properties

- *A invertible $\Rightarrow \mathcal{R}^A$ is a permutation matrix*
- *A is square with generic rank profile $\Rightarrow \mathcal{R}^A = I_n$*
- *RowRP(A) = RowSupport$(\mathcal{R}^A)$*
- *ColRP(A) = ColSupport$(\mathcal{R}^A)$*

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{matrix} \text{RowRP} = \{1,3,4\} \\ \text{ColRP} = \{1,2,4\} \end{matrix} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Properties of the rank profile matrix

Properties

- A *invertible* $\Rightarrow \mathcal{R}^A$ *is a permutation matrix*
- A *is square with generic rank profile* $\Rightarrow \mathcal{R}^A = I_n$
- *RowRP*(A) = *RowSupport*$(\mathcal{R}^A)$
- *ColRP*(A) = *ColSupport*$(\mathcal{R}^A)$
- *RowRP*$(A_{1..i,1..j})$ = *RowSupport*$({\mathcal{R}^A}_{1..i,1..j})$
- *ColRP*$(A_{1..i,1..j})$ = *ColSupport*$({\mathcal{R}^A}_{1..i,1..j})$

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ 2 & 5 & 4 & 7 \end{bmatrix} \qquad \begin{array}{c} \text{RowRP} = \{1,3\} \\ \text{ColRP} = \{1,2\} \end{array} \qquad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

When does a PLUQ decomposition reveal the rank profile matrix ?

Focus on the pivoting strategy:

- Pivot search
- Permutation to bring the pivot to the main diagonal

When does a PLUQ decomposition reveal the rank profile matrix ?

Focus on the pivoting strategy:

- Pivot search
- Permutation to bring the pivot to the main diagonal

$$A = PLUQ = P \begin{bmatrix} L & 0 \\ M & I_{m-r} \end{bmatrix} \qquad \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} \qquad \begin{bmatrix} U & V \\ & I_{n-r} \end{bmatrix} Q$$

When does a PLUQ decomposition reveal the rank profile matrix ?

Focus on the pivoting strategy:

- Pivot search
- Permutation to bring the pivot to the main diagonal

$$A = PLUQ = P \begin{bmatrix} L & 0 \\ M & I_{m-r} \end{bmatrix} \quad \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} \quad \begin{bmatrix} U & V \\ & I_{n-r} \end{bmatrix} Q$$

When does a PLUQ decomposition reveal the rank profile matrix ?

Focus on the pivoting strategy:

- Pivot search
- Permutation to bring the pivot to the main diagonal

$$A = PLUQ = P\begin{bmatrix} L & 0 \\ M & I_{m-r} \end{bmatrix} P^T P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q Q^T \begin{bmatrix} U & V \\ & I_{n-r} \end{bmatrix} Q$$

When does a PLUQ decomposition reveal the rank profile matrix ?

Focus on the pivoting strategy:

- Pivot search
- Permutation to bring the pivot to the main diagonal

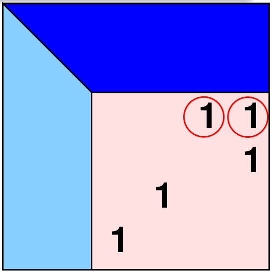
$$A = PLUQ = \underbrace{P \begin{bmatrix} L & 0 \\ M & I_{m-r} \end{bmatrix} P^T}_{\overline{L}} \underbrace{P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q}_{\mathcal{R}^A} \underbrace{Q^T \begin{bmatrix} U & V \\ & I_{n-r} \end{bmatrix} Q}_{\overline{U}}$$

Pivoting and permutation strategies

Pivot Search

Pivot's (i, j) position minimizes some pre-order:

Row order: any non-zero on the first non-zero row

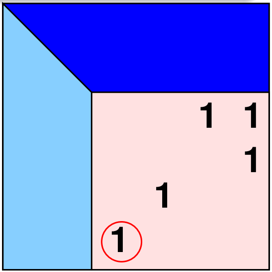


Pivoting and permutation strategies

Pivot Search

Pivot's (i, j) position minimizes some pre-order:

Row/Col order: any non-zero on the first non-zero row/col



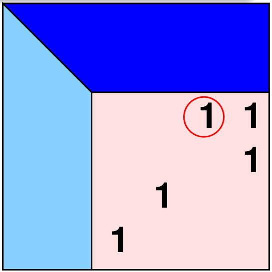
Pivoting and permutation strategies

Pivot Search

Pivot's (i, j) position minimizes some pre-order:

Row/Col order: any non-zero on the first non-zero row/col

Lex order: first non-zero on the first non-zero row



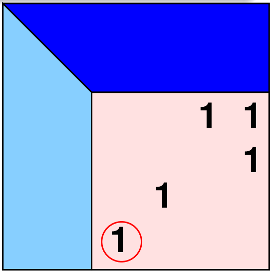
Pivoting and permutation strategies

Pivot Search

Pivot's (i, j) position minimizes some pre-order:

Row/Col order: any non-zero on the first non-zero row/col

Lex/RevLex order: first non-zero on the first non-zero row/col



Pivoting and permutation strategies

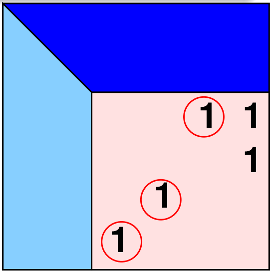
Pivot Search

Pivot's (i, j) position minimizes some pre-order:

Row/Col order: any non-zero on the first non-zero row/col

Lex/RevLex order: first non-zero on the first non-zero row/col

Product order: first non-zero in the (i, j) leading sub-matrix



Pivoting and permutation strategies

Pivot Search

Pivot's (i, j) position minimizes some pre-order:

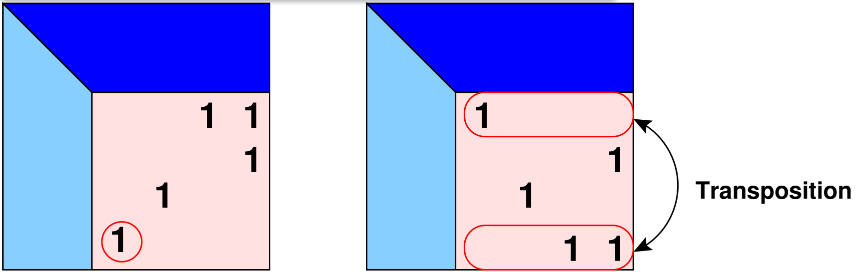
Row/Col order: any non-zero on the first non-zero row/col

Lex/RevLex order: first non-zero on the first non-zero row/col

Product order: first non-zero in the (i, j) leading sub-matrix

Permutation

- Transpositions



Pivoting and permutation strategies

Pivot Search

Pivot's (i,j) position minimizes some pre-order:

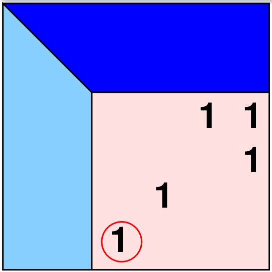
Row/Col order: any non-zero on the first non-zero row/col

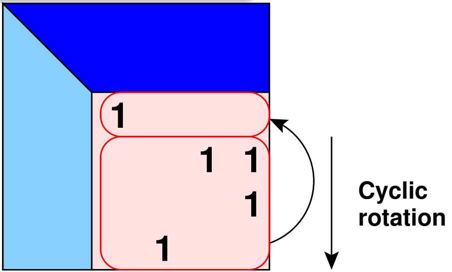
Lex/RevLex order: first non-zero on the first non-zero row/col

Product order: first non-zero in the (i,j) leading sub-matrix

Permutation

- Transpositions
- Cyclic Rotations





Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

Search	**Row perm.**	**Col. perm.**	RowRP	ColRP	$\mathcal{R}^A$	**Instance**
Row order Col. order						
Lexico.						
Rev. lex.						
Product						

Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

Search	Row perm.	Col. perm.	RowRP	ColRP	$\mathcal{R}^A$	Instance
Row order Col. order	Transposition	Transposition	✓			[IMH82] [JPS13]
Lexico.						
Rev. lex.						
Product						

- RowRP $= \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$

Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

Search	**Row perm.**	**Col. perm.**	RowRP	ColRP	$\mathcal{R}^A$	**Instance**
Row order	Transposition	Transposition	✓			[IMH82] [JPS13]
Col. order	Transposition	Transposition		✓		[KG85] [JPS13]
Lexico.						
Rev. lex.						
Product						

- RowRP $= [\,1\ 2\ \dots\ m\,]\, P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$
- ColRP $= [\,I_r\ 0\,]\, Q\, [\,1\ 2\ \dots\ m\,]^T$

Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

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]
Rev. lex.	Transposition	Transposition		✓		[Sto00]
Product						

- RowRP $= [1\ 2\ \ldots\ m]\, P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$
- ColRP $= [I_r\ 0]\, Q\, [1\ 2\ \ldots\ m]^T$

Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

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]
Rev. lex.	Transposition	Transposition		✓		[Sto00]
Product	Rotation	Rotation	✓	✓	✓	[DPS13]

- RowRP $= \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$
- ColRP $= \begin{bmatrix} I_r & 0 \end{bmatrix} Q \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix}^T$
- $\mathcal{R}^A = P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q$

Pivoting strategies revealing rank profiles

For any type of PLUQ algorithm: iterative / block iterative / recursive

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]
Rev. lex.	Transposition	Transposition		✓		[Sto00]
Product	Rotation	Transposition	✓			[DPS15]
Product	Transposition	Rotation		✓		[DPS15]
Product	Rotation	Rotation	✓	✓	✓	[DPS13]

- RowRP $= \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$
- ColRP $= \begin{bmatrix} I_r & 0 \end{bmatrix} Q \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix}^T$
- $\mathcal{R}^A = P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q$

Pivoting strategies revealing rank profiles

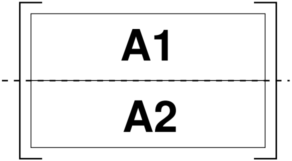
For any type of PLUQ algorithm: iterative / block iterative / recursive

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]

- RowRP $= \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} P \begin{bmatrix} I_r \\ 0 \end{bmatrix}$
- ColRP $= \begin{bmatrix} I_r & 0 \end{bmatrix} Q \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix}^T$
- $\mathcal{R}^A = P \begin{bmatrix} I_r & \\ & 0 \end{bmatrix} Q$

The slab recursive algorithm

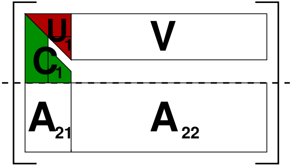
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise

The slab recursive algorithm

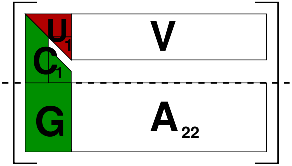
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise
2. Recursive call on A_1

The slab recursive algorithm

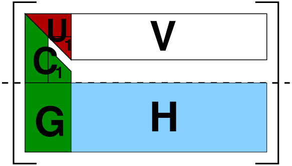
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ (trsm)

The slab recursive algorithm

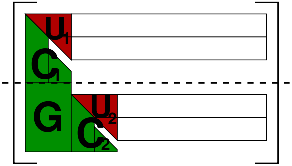
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ (trsm)
4. $H \leftarrow A_{22} - G \times V$ (MM)

The slab recursive algorithm

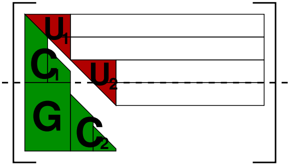
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ (trsm)
4. $H \leftarrow A_{22} - G \times V$ (MM)
5. Recursive call on H

The slab recursive algorithm

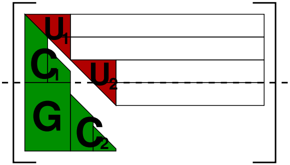
Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ (trsm)
4. $H \leftarrow A_{22} - G \times V$ (MM)
5. Recursive call on H
6. Row permutations

The slab recursive algorithm

Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



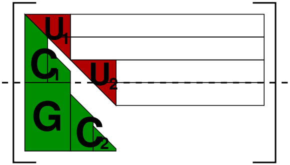
1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ ($\texttt{trsm}$)
4. $H \leftarrow A_{22} - G \times V$ ($\texttt{MM}$)
5. Recursive call on H
6. Row permutations

Implements the lexicographic order search.

- Col/Row Transpositions : Computes the ColRP

The slab recursive algorithm

Slab Recursive LU [IMH82, KG85, Sto00, JPS13]



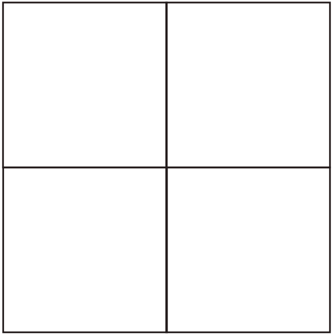
1. Split A Row-wise
2. Recursive call on A_1
3. $G \leftarrow A_{21}U_1^{-1}$ ($\texttt{trsm}$)
4. $H \leftarrow A_{22} - G \times V$ ($\texttt{MM}$)
5. Recursive call on H
6. Row permutations

Implements the lexicographic order search.

- Col/Row Transpositions : Computes the ColRP
- Row Rotations : Computes $\mathcal{R}^A$ [DPS15]

The tiled recursive algorithm

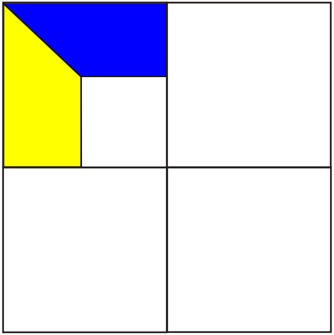
[DPS13]



2×2 block splitting

The tiled recursive algorithm

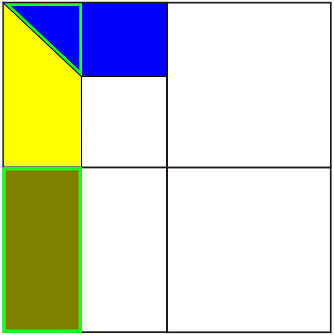
[DPS13]



Recursive call

The tiled recursive algorithm

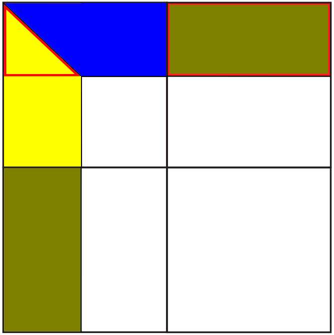
[DPS13]



TRSM: $B \leftarrow BU^{-1}$

The tiled recursive algorithm

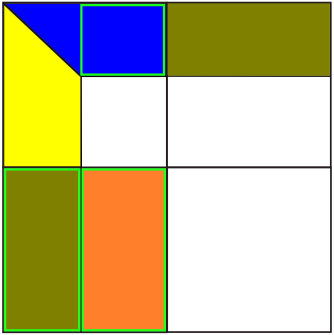
[DPS13]



TRSM: $B \leftarrow L^{-1}B$

The tiled recursive algorithm

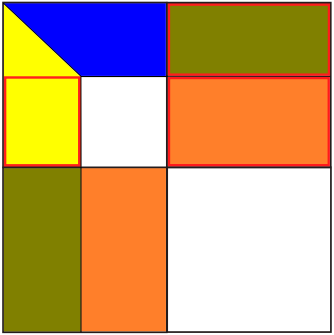
[DPS13]



MatMul: $C \leftarrow C - A \times B$

The tiled recursive algorithm

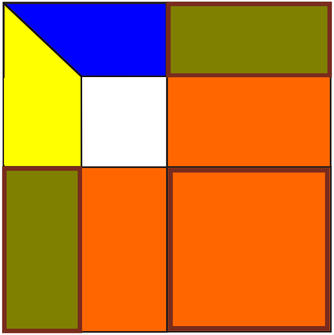
[DPS13]



`MatMul`: $C \leftarrow C - A \times B$

The tiled recursive algorithm

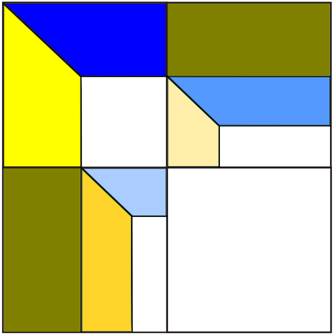
[DPS13]



`MatMul`: $C \leftarrow C - A \times B$

The tiled recursive algorithm

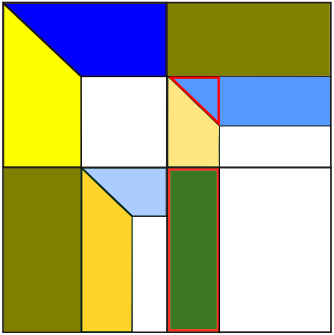
[DPS13]



2 independent recursive calls (compatible with the **product order**)

The tiled recursive algorithm

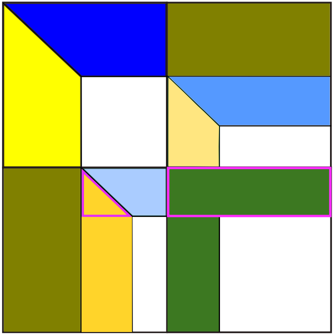
[DPS13]



TRSM: $B \leftarrow BU^{-1}$

The tiled recursive algorithm

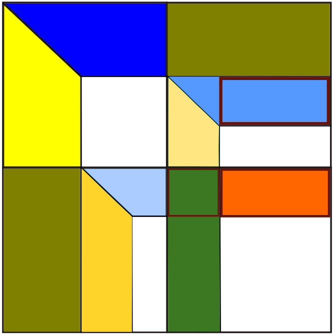
[DPS13]



TRSM: $B \leftarrow L^{-1}B$

The tiled recursive algorithm

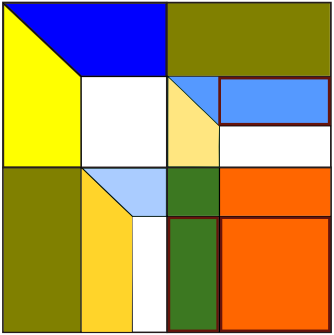
[DPS13]



`MatMul`: $C \leftarrow C - A \times B$

The tiled recursive algorithm

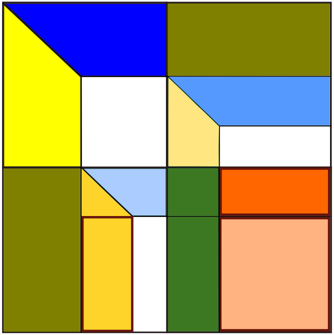
[DPS13]



`MatMul`: $C \leftarrow C - A \times B$

The tiled recursive algorithm

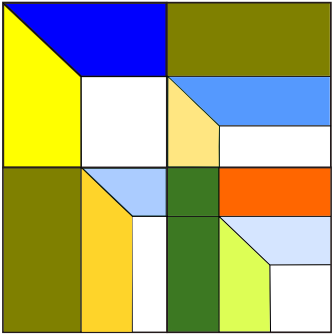
[DPS13]



MatMul: $C \leftarrow C - A \times B$

The tiled recursive algorithm

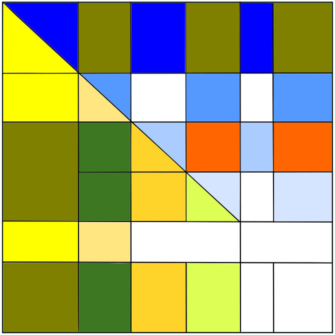
[DPS13]



Recursive call

The tiled recursive algorithm

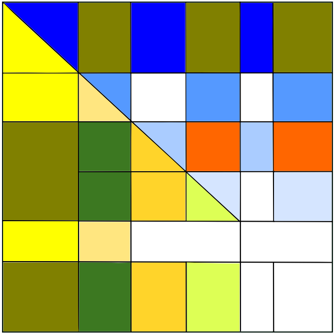
[DPS13]



Puzzle game (block **rotations**)

The tiled recursive algorithm

[DPS13]



- $O(mnr^{\omega-2})$ ($2/3n^3$ for $\omega = 3$)
- fewer modular reductions than slab algorithms
- rank deficiency introduces parallelism

Outline

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n \times$ Lower, Upper $\times J_n$

$$\begin{bmatrix} 2 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ 6 & 6 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ 3 & 2 & 2 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ 6 & 6 & 2 & 2 & 5 & 1 & 4 & 5 & 5 & 2 \\ 0 & 1 & 6 & 2 & 4 & 4 & 5 & 0 & 3 & 3 \\ 2 & 0 & 5 & 6 & 3 & 1 & 0 & 6 & 2 & 4 \\ 6 & 2 & 6 & 1 & 3 & 4 & 6 & 5 & 2 & 6 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 4 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 2 & 3 \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 6 \end{bmatrix}$$

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n \times$ Lower, Upper $\times J_n$

$$\begin{bmatrix} 0 &&&&&&&&& \\ 6 & 0 &&&&&&&& \\ 3 & 2 & 0 &&&&&&& \\ 6 & 6 & 2 & 0 &&&&&& \\ 0 & 1 & 6 & 2 & 0 &&&&& \\ 2 & 0 & 5 & 6 & 3 & 0 &&&& \\ 6 & 2 & 6 & 1 & 3 & 4 & 0 &&& \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 0 && \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 0 & \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 6 & 1 & 0 & 1 & 4 & 6 & 5 & 2 & 6 \\ & 0 & 2 & 5 & 6 & 1 & 6 & 1 & 3 & 1 \\ && 0 & 6 & 4 & 1 & 2 & 2 & 0 & 3 \\ &&& 0 & 5 & 1 & 4 & 5 & 5 & 2 \\ &&&& 0 & 4 & 5 & 0 & 3 & 3 \\ &&&&& 0 & 0 & 6 & 2 & 4 \\ &&&&&& 0 & 5 & 2 & 6 \\ &&&&&&& 0 & 5 & 4 \\ &&&&&&&& 0 & 3 \\ &&&&&&&&& 0 \end{bmatrix}$$

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n \times$ Lower, Upper $\times J_n$

$$\begin{bmatrix} 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 0 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 0 & \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 0 & & \\ 6 & 2 & 6 & 1 & 3 & 4 & 0 & & & \\ 2 & 0 & 5 & 6 & 3 & 0 & & & & \\ 0 & 1 & 6 & 2 & 0 & & & & & \\ 6 & 6 & 2 & 0 & & & & & & \\ 3 & 2 & 0 & & & & & & & \\ 6 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}, \begin{bmatrix} 6 & 2 & 5 & 6 & 4 & 1 & 0 & 1 & 6 & 0 \\ 1 & 3 & 1 & 6 & 1 & 6 & 5 & 2 & 0 & \\ 3 & 0 & 2 & 2 & 1 & 4 & 6 & 0 & & \\ 2 & 5 & 5 & 4 & 1 & 5 & 0 & & & \\ 3 & 3 & 0 & 5 & 4 & 0 & & & & \\ 4 & 2 & 6 & 0 & 0 & & & & & \\ 6 & 2 & 5 & 0 & & & & & & \\ 4 & 5 & 0 & & & & & & & \\ 3 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}$$

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n\times$ Lower, Upper $\times J_n$

$$\begin{bmatrix} 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 0 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 0 & \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 0 & & \\ 6 & 2 & 6 & 1 & 3 & 4 & 0 & & & \\ 2 & 0 & 5 & 6 & 3 & 0 & & & & \\ 0 & 1 & 6 & 2 & 0 & & & & & \\ 6 & 6 & 2 & 0 & & & & & & \\ 3 & 2 & 0 & & & & & & & \\ 6 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}, \begin{bmatrix} 6 & 2 & 5 & 6 & 4 & 1 & 0 & 1 & 6 & 0 \\ 1 & 3 & 1 & 6 & 1 & 6 & 5 & 2 & 0 & \\ 3 & 0 & 2 & 2 & 1 & 4 & 6 & 0 & & \\ 2 & 5 & 5 & 4 & 1 & 5 & 0 & & & \\ 3 & 3 & 0 & 5 & 4 & 0 & & & & \\ 4 & 2 & 6 & 0 & 0 & & & & & \\ 6 & 2 & 5 & 0 & & & & & & \\ 4 & 5 & 0 & & & & & & & \\ 3 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}$$

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n\times$ Lower, Upper $\times J_n$

$$\begin{bmatrix} 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 0 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 0 & \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 0 & & \\ 6 & 2 & 6 & 1 & 3 & 4 & 0 & & & \\ 2 & 0 & 5 & 6 & 3 & 0 & & & & \\ 0 & 1 & 6 & 2 & 0 & & & & & \\ 6 & 6 & 2 & 0 & & & & & & \\ 3 & 2 & 0 & & & & & & & \\ 6 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}, \begin{bmatrix} 6 & 2 & 5 & 6 & 4 & 1 & 0 & 1 & 6 & 0 \\ 1 & 3 & 1 & 6 & 1 & 6 & 5 & 2 & 0 & \\ 3 & 0 & 2 & 2 & 1 & 4 & 6 & 0 & & \\ 2 & 5 & 5 & 4 & 1 & 5 & 0 & & & \\ 3 & 3 & 0 & 5 & 4 & 0 & & & & \\ 4 & 2 & 6 & 0 & 0 & & & & & \\ 6 & 2 & 5 & 0 & & & & & & \\ 4 & 5 & 0 & & & & & & & \\ 3 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}$$

Finding the quasiseparability orders

Left triangular matrices

Of the form: $\begin{bmatrix} * & * & * & * & 0 \\ * & * & * & 0 & \\ * & * & 0 & & \\ * & 0 & & & \\ 0 & & & & \end{bmatrix}$

Examples: $J_n \times$ Lower, Upper $\times J_n$

$$\left[\begin{array}{cccccccccc} 1 & 1 & 5 & 5 & 2 & 1 & 3 & 4 & 5 & 0 \\ 0 & 1 & 6 & 2 & 4 & 4 & 3 & 3 & 0 & \\ 1 & 1 & 5 & 5 & 2 & 1 & 3 & 0 & & \\ 6 & 2 & 6 & 1 & 3 & 4 & 0 & & & \\ 2 & 0 & 5 & 6 & 3 & 0 & & & & \\ 0 & 1 & 6 & 2 & 0 & & & & & \\ 6 & 6 & 2 & 0 & & & & & & \\ 3 & 2 & 0 & & & & & & & \\ 6 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{array}\right], \left[\begin{array}{cccccccccc} 6 & 2 & 5 & 6 & 4 & 1 & 0 & 1 & 6 & 0 \\ 1 & 3 & 1 & 6 & 1 & 6 & 5 & 2 & 0 & \\ 3 & 0 & 2 & 2 & 1 & 4 & 6 & 0 & & \\ 2 & 5 & 5 & 4 & 1 & 5 & 0 & & & \\ 3 & 3 & 0 & 5 & 4 & 0 & & & & \\ 4 & 2 & 6 & 0 & 0 & & & & & \\ 6 & 2 & 5 & 0 & & & & & & \\ 4 & 5 & 0 & & & & & & & \\ 3 & 0 & & & & & & & & \\ 0 & & & & & & & & & \end{array}\right]$$

Quasiseparability orders on the rank profile matrix

$$
\begin{bmatrix}
2 & 2 & 1 & 0 & 2 & 1 & 1 & 1 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 2 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0\\
2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & & & \\
0 & 0 & 0 & 0 & 1 & & & & & \\
0 & 1 & 0 & 0 & & & & & & \\
0 & 0 & 0 & & & & & & & \\
0 & 0 & 1 & & & & & & & \\
0 & & & & & & & & &
\end{bmatrix}
$$

Rank $= 1$ Quasiseparability order $= 1$

Quasiseparability orders on the rank profile matrix

$$
\begin{bmatrix}
2 & 2 & 1 & 0 & 2 & 1 & 1 & 1 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 2 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\
2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & & & \\
0 & 0 & 0 & 0 & 1 & & & & & \\
0 & 1 & 0 & 0 & & & & & & \\
0 & 0 & 0 & & & & & & & \\
0 & 0 & 1 & & & & & & & \\
0 & & & & & & & & &
\end{bmatrix}
$$

Rank $= 2$ Quasiseparability order $= 2$

Quasiseparability orders on the rank profile matrix

$$\begin{bmatrix} 2 & 2 & 1 & 0 & 2 & 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 0 & 0 \\ 2 & 2 & 1 & 0 & 2 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & & & \\ 0 & 0 & 0 & 0 & 1 & & & & & \\ 0 & 1 & 0 & 0 & & & & & & \\ 0 & 0 & 0 & & & & & & & \\ 0 & 0 & 1 & & & & & & & \\ 0 & & & & & & & & & \end{bmatrix}$$

Rank $= 2$ Quasiseparability order $= 2$

Quasiseparability orders on the rank profile matrix

$$
\begin{bmatrix}
2 & 2 & 1 & 0 & 2 & 1 & 1 & 1 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 2 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\
2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & & & \\
0 & 0 & 0 & 0 & 1 & & & & & \\
0 & 1 & 0 & 0 & & & & & & \\
0 & 0 & 0 & & & & & & & \\
0 & 0 & 1 & & & & & & & \\
0 & & & & & & & & &
\end{bmatrix}
$$

Rank $= 1$ Quasiseparability order $= 2$

Quasiseparability orders on the rank profile matrix

$$\begin{bmatrix} 2&2&1&0&2&1&1&1&0&0\\ 2&2&1&0&2&0&1&2&0&0\\ 2&2&1&0&2&2&1&0&0&0\\ 0&0&0&0&0&0&0&0&0&0\\ 2&2&1&0&2&0&0&0&0&0\\ 2&2&1&0&0&0&0&0&0&0\\ 2&1&0&0&0&0&0&0&0&0\\ 1&2&0&0&0&0&0&0&0&0\\ 2&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&\\ 0&0&0&0&0&0&0&0&&\\ 0&0&0&0&0&0&0&&&\\ 0&0&0&0&0&0&1&&&\\ 0&0&0&0&1&&&&&\\ 0&1&0&0&&&&&&\\ 0&0&0&&&&&&&\\ 0&0&1&&&&&&&\\ 0&&&&&&&&& \end{bmatrix}$$

Rank $= 1$ Quasiseparability order $= 2$

Major difficulty

Complexity in $O(n^2 r^{\omega-2})$ where $r = \text{rank}(A) \gg r_L, r_U$. But

Quasiseparability orders on the rank profile matrix

$$
\begin{bmatrix}
2 & 2 & 1 & 0 & 2 & 1 & 1 & 1 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 2 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0\\
2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & & & \\
0 & 0 & 0 & 0 & 1 & & & & & \\
0 & 1 & 0 & 0 & & & & & & \\
0 & 0 & 0 & & & & & & & \\
0 & 0 & 1 & & & & & & & \\
0 & & & & & & & & &
\end{bmatrix}
$$

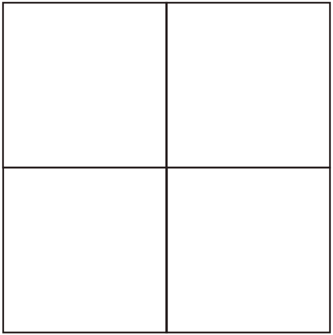
Rank $= 1$ Quasiseparability order $= 2$

Major difficulty

Complexity in $O(n^2 r^{\omega-2})$ where $r = \text{rank}(A) \gg r_L, r_U$. But

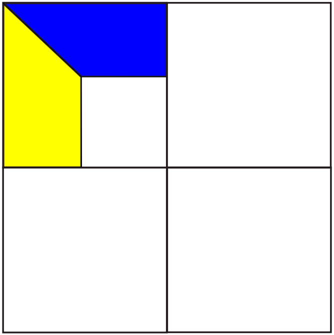
- only a few pivots ($O(r_L, r_U)$) near the top left corner
- numerous pivots lie near the anti-diagonal $\rightsquigarrow$ cheaper elimination

Computing the left-triangular part of a rank profile matrix



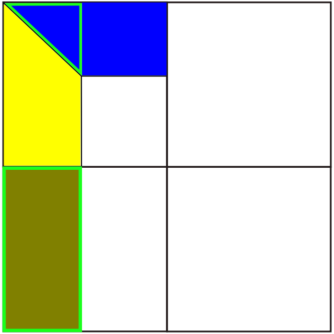
2×2 block splitting

Computing the left-triangular part of a rank profile matrix



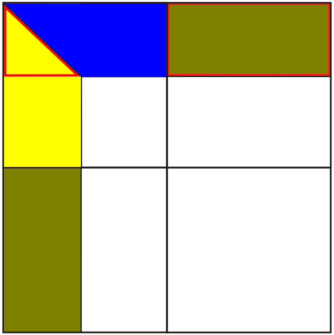
Recursive call

Computing the left-triangular part of a rank profile matrix



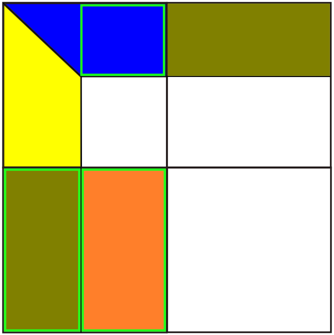
TRSM: $B \leftarrow BU^{-1}$

Computing the left-triangular part of a rank profile matrix



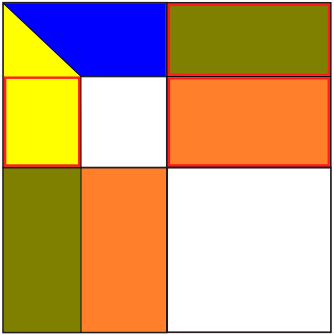
TRSM: $B \leftarrow L^{-1}B$

Computing the left-triangular part of a rank profile matrix



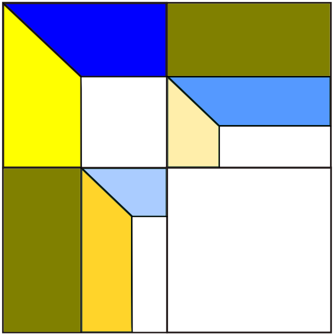
`MatMul`: $C \leftarrow C - A \times B$

Computing the left-triangular part of a rank profile matrix



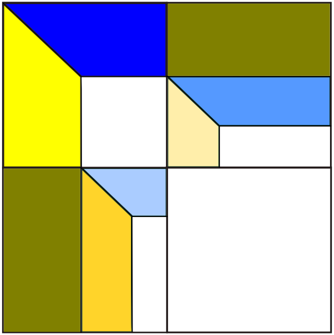
MatMul: $C \leftarrow C - A \times B$

Computing the left-triangular part of a rank profile matrix



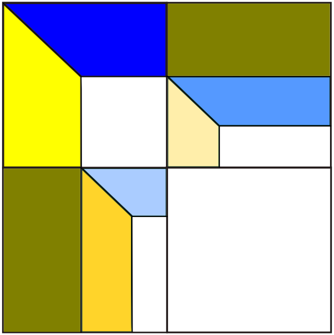
2 recursive calls : not possible as no longer left-triangular

Computing the left-triangular part of a rank profile matrix



2 recursive calls : not possible as no longer left-triangular
$\rightsquigarrow$ clear out and permute back to original position beforehand

Computing the left-triangular part of a rank profile matrix



2 recursive calls : not possible as no longer left-triangular

$\rightsquigarrow$ clear out and permute back to original position beforehand

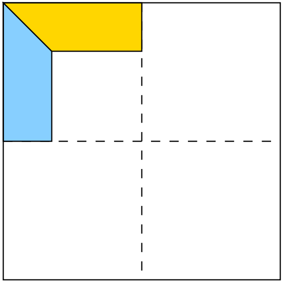
- $O(n^2 s^{\omega-2})$
- in place

Outline

The binary tree of PLUQ generator

Principle: *à la* Hierarchically Semiseparable representations

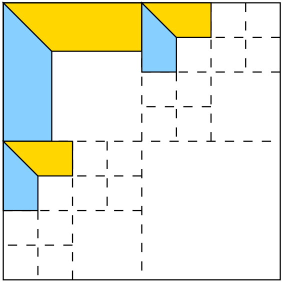
- Split in 4 quadrants
- Store the top left corner as a PLUQ decomposition: $O(nr)$



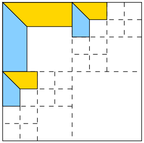
The binary tree of PLUQ generator

Principle: *à la* Hierarchically Semiseparable representations

- Split in 4 quadrants
- Store the top left corner as a PLUQ decomposition: $O(nr)$
- Apply recursively on the anti-diagonal blocks



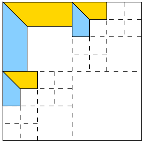
The binary tree of PLUQ generator



For a quasiseparability order s:

- $s \geq$ all ranks of the PLUQ

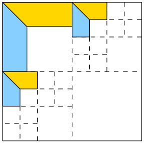
The binary tree of PLUQ generator



For a quasiseparability order s:

- $s \geq$ all ranks of the PLUQ
- ✓Size: $\sum_{i=1}^{\log n/s} 2^{i-1} \frac{n}{2^i} s = O(ns \log \frac{n}{s})$.

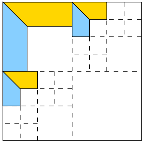
The binary tree of PLUQ generator



For a quasiseparability order s:

- $s \geq$ all ranks of the PLUQ
- ✓Size: $\sum_{i=1}^{\log n/s} 2^{i-1} \frac{n}{2^i} s = O(ns \log \frac{n}{s})$.
- ✓Computed in: $\sum_{i=1}^{\log n/s} 2^{i-1} \left(\frac{n}{2^i}\right)^2 s^{\omega-2} = O(n^2 s^{\omega-2})$.

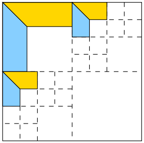
The binary tree of PLUQ generator



For a quasiseparability order s:

- $s \geq$ all ranks of the PLUQ
- ✓Size: $\sum_{i=1}^{\log n/s} 2^{i-1} \frac{n}{2^i} s = O(ns \log \frac{n}{s})$.
- ✓Computed in: $\sum_{i=1}^{\log n/s} 2^{i-1} \left(\frac{n}{2^i}\right)^2 s^{\omega-2} = O(n^2 s^{\omega-2})$.
- ✗No mathematical reconstruction formula

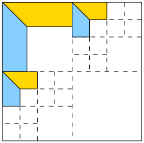
The binary tree of PLUQ generator



For a quasiseparability order s:

- $s \geq$ all ranks of the PLUQ
- ✓Size: $\sum_{i=1}^{\log n/s} 2^{i-1} \frac{n}{2^i} s = O(ns \log \frac{n}{s})$.
- ✓Computed in: $\sum_{i=1}^{\log n/s} 2^{i-1} \left(\frac{n}{2^i}\right)^2 s^{\omega-2} = O(n^2 s^{\omega-2})$.
- ✗No mathematical reconstruction formula
- ✓QuasiSep $\times$ Vector in $O(\text{Size}) = O(ns \log \frac{n}{s})$

The binary tree of PLUQ generator



For a quasiseparability order s:

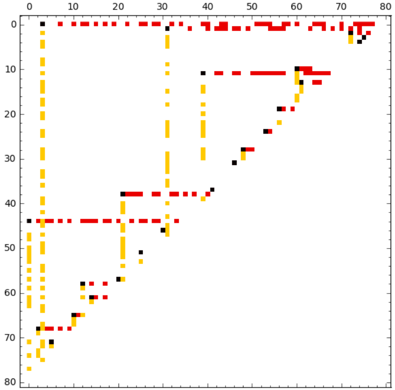
- $s \geq$ all ranks of the PLUQ
- ✓Size: $\sum_{i=1}^{\log n/s} 2^{i-1} \frac{n}{2^i} s = O(ns \log \frac{n}{s})$.
- ✓Computed in: $\sum_{i=1}^{\log n/s} 2^{i-1} \left(\frac{n}{2^i}\right)^2 s^{\omega-2} = O(n^2 s^{\omega-2})$.
- ✗No mathematical reconstruction formula
- ✓QuasiSep × Vector in $O(\text{Size}) = O(ns \log \frac{n}{s})$
- ✓QuasiSep × QuasiSep in $O(n^2 s^{\omega-2})$

The Bruhat generator

Idea: *There can not be too many pivots near and far from the anti-diagonal.*

From a PLUQ decomposition revealing the rank profile matrix E:

$$\mathcal{L} = \text{LeftTriang}(P \begin{bmatrix} L & 0 \end{bmatrix} Q),$$
$$\mathcal{E} = \text{LeftTriang}(E),$$
$$\mathcal{U} = \text{LeftTriang}(P \begin{bmatrix} U \\ 0 \end{bmatrix} Q).$$



The Bruhat Generator

For a quasiseparability order s:

- ✓Size: $O(ns)$

Lemma

There is less than ns non zero elements in $\mathcal{L}$.

Proof.

$\mathcal{L}$ is left triangular of QS order s.
The k-th row of $\mathcal{L}$ has $\leq s$ pivots above $\rightsquigarrow \leq s$ non-zeros. □

The Bruhat Generator

For a quasiseparability order s:

- ✓Size: $O(ns)$
- ✓Generator: $A = \text{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$

$$A = P \begin{bmatrix} L & 0 \end{bmatrix} Q Q^T \begin{bmatrix} U \\ 0 \end{bmatrix} Q$$

The Bruhat Generator

For a quasiseparability order s:

- ✓Size: $O(ns)$
- ✓Generator: $A = \text{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$

$$\mathcal{A} = \text{LeftTriang}(\mathcal{L}Q^T \begin{bmatrix} U \\ 0 \end{bmatrix} Q)$$

Lemma

If $A = BU$ with U upper triangular, then
$\text{LeftTriang}(A) = \text{LeftTriang}(\text{LeftTriang}(B)U)$

The Bruhat Generator

For a quasiseparability order s:

- ✓Size: $O(ns)$
- ✓Generator: $A = \mathsf{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$

$$\mathcal{A} = \mathsf{LeftTriang}(\mathcal{L}E^T P \begin{bmatrix} U \\ 0 \end{bmatrix} Q)$$

$$E = P \begin{bmatrix} I_r \\ 0 \end{bmatrix} Q$$

The Bruhat Generator

For a quasiseparability order s:

- ✓Size: $O(ns)$
- ✓Generator: $A = \text{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$

$$\mathcal{A} = \text{LeftTriang}(\mathcal{L}E^T\mathcal{U})$$

Lemma

If $A = LB$ with L lower triangular, then
$\textit{LeftTriang}(A) = \textit{LeftTriang}(L\,\textit{LeftTriang}(B))$

The Bruhat Generator

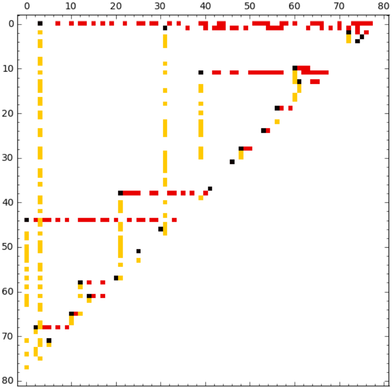
For a quasiseparability order s:

- ✓Size: $O(ns)$
- ✓Generator: $A = \text{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$

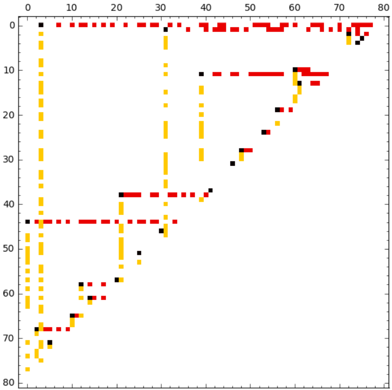
$$\mathcal{A} = \text{LeftTriang}(\mathcal{L}\mathcal{E}^T\mathcal{U})$$

Lemma

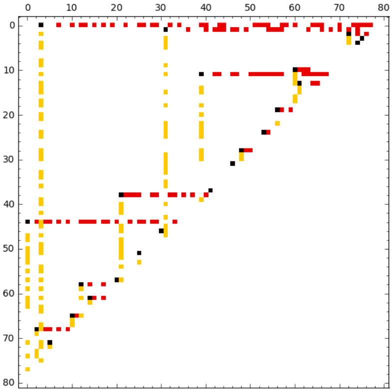
If $A = LB$ with L lower triangular, then
$\mathit{LeftTriang}(A) = \mathit{LeftTriang}(L\,\mathit{LeftTriang}(B))$



- ✓Computed in $O(n^2 s^{\omega-2})$
  ⇝ Adapted from the tile recursive PLUQ algo

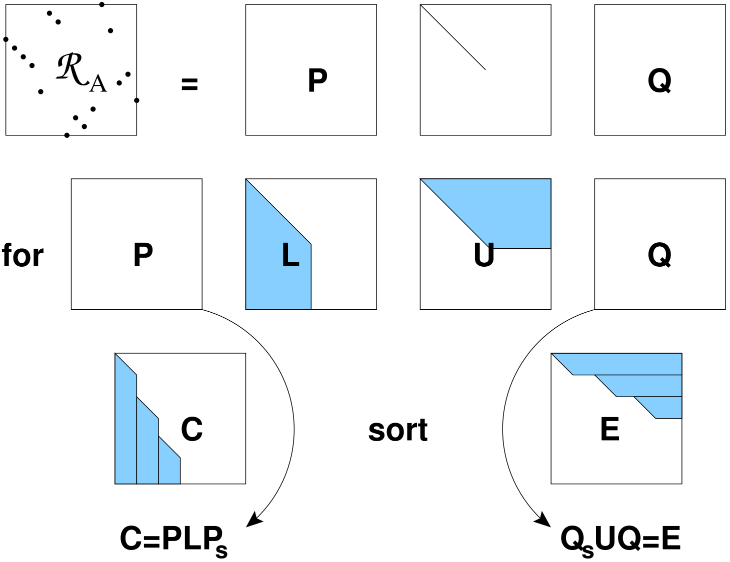


- ✓Computed in $O(n^2 s^{\omega-2})$
  $\rightsquigarrow$ Adapted from the tile recursive PLUQ algo
- ✓QuasiSep $\times$ Vector in $O(\text{Size}) = O(ns)$



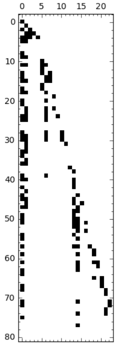
- ✓Computed in $O(n^2 s^{\omega-2})$
  ⇝ Adapted from the tile recursive PLUQ algo
- ✓QuasiSep $\times$ Vector in $O(\text{Size}) = O(ns)$
- ✗Scattered⇝ no fast QuasiSep $\times$ QuasiSep

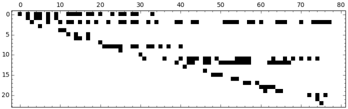
Row and column echelon forms from PLUQ



The compact Bruhat Generator

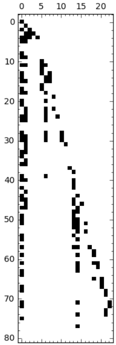
- $\mathcal{C} = \mathcal{LQ}$: Col. permutation to column echelon form
- $\mathcal{R} = \mathcal{PU}$: Row permutation to row echelon form
- $A = \text{LeftTriang}(\mathcal{CSR})$ for an $r \times r$ permutation S (generalized Bruhat decomposition[MH07]).

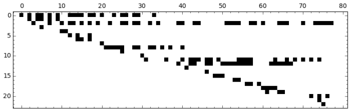




Major difficulty

Again $r = \text{rank}(A) \gg s$

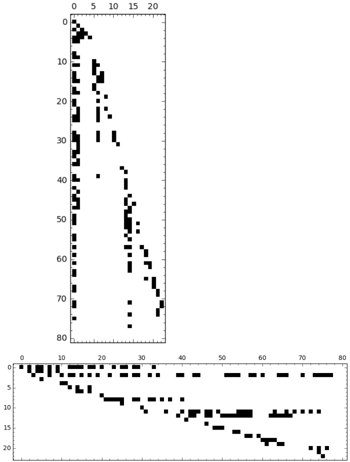




Major difficulty

Again $r = \text{rank}(A) \gg s$

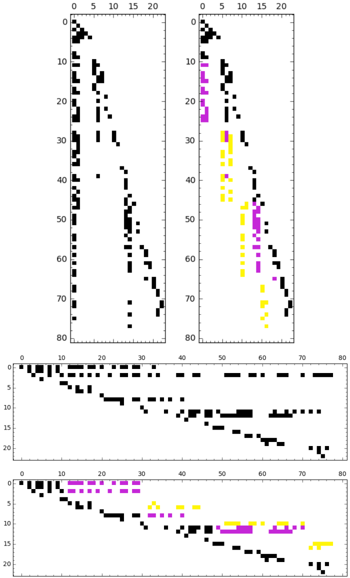
- But double structure: either echelon or Left-triangular



Major difficulty

Again $r = \text{rank}(A) \gg s$

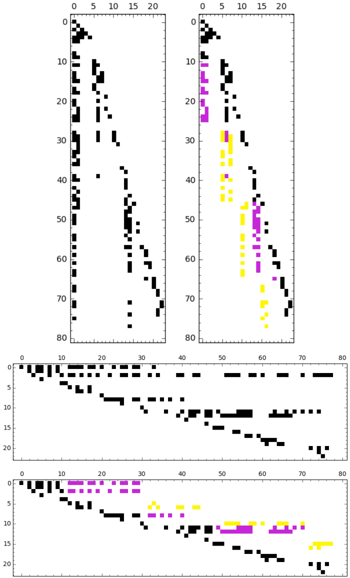
- But double structure: either echelon or Left-triangular
- Fold them in blocks of size s



Major difficulty

Again $r = \text{rank}(A) \gg s$

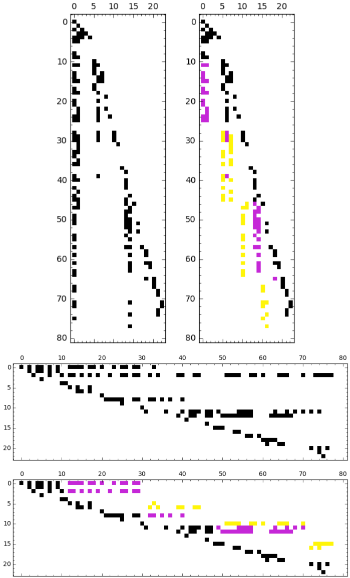
- But double structure: either echelon or Left-triangular
- Fold them in blocks of size s
- forming a **dense** $O(ns)$ block bi-diagonal structure



Major difficulty

Again $r = \text{rank}(A) \gg s$

- But double structure: either echelon or Left-triangular
- Fold them in blocks of size s
- forming a **dense** $O(ns)$ block bi-diagonal structure
- QuasiSep $\times$ QuasiSep in $O(n^2 s^{\omega-2})$



Multiplying two quasiseparable matrices

Left triangular × Left triangular

$$A \times B = C_A R_A E_A \times C_B R_B E_B.$$

Multiplying two quasiseparable matrices

Left triangular × Left triangular

$$A \times B = C_A(R_A(E_A \times C_B)R_B)E_B.$$

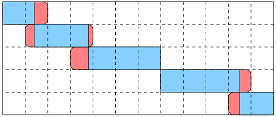
$$E_A \times C_B = (D_{\mathcal{U}_A} + T_{\mathcal{U}_A} S_{\mathcal{U}_A})(D_{\mathcal{L}_B} + S_{\mathcal{L}_B} T_{\mathcal{L}_B})$$

Multiplying two quasiseparable matrices

Left triangular $\times$ Left triangular

$$\mathsf{A} \times \mathsf{B} = \mathsf{C_A}(\mathsf{R_A}(\mathsf{E_A} \times \mathsf{C_B})\mathsf{R_B})\mathsf{E_B}.$$

$\mathsf{E_A} \times \mathsf{C_B} = (\mathsf{D}_{\mathcal{U}_\mathsf{A}} + \mathsf{T}_{\mathcal{U}_\mathsf{A}}\mathsf{S}_{\mathcal{U}_\mathsf{A}})(\mathsf{D}_{\mathcal{L}_\mathsf{B}} + \mathsf{S}_{\mathcal{L}_\mathsf{B}}\mathsf{T}_{\mathcal{L}_\mathsf{B}})$
$\rightsquigarrow$ products of block (sub)-diagonal $r_A \times n$ and $r_B \times n$



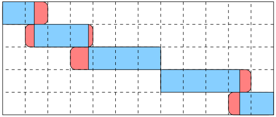
$\rightsquigarrow O(s^{\omega-1}n)$

Multiplying two quasiseparable matrices

Left triangular × Left triangular

$$\mathsf{A} \times \mathsf{B} = \mathsf{C_A}(\mathsf{R_A}(\mathsf{E_A} \times \mathsf{C_B})\mathsf{R_B})\mathsf{E_B}.$$

$\mathsf{E_A} \times \mathsf{C_B} = (\mathsf{D}_{\mathcal{U}_\mathsf{A}} + \mathsf{T}_{\mathcal{U}_\mathsf{A}}\mathsf{S}_{\mathcal{U}_\mathsf{A}})(\mathsf{D}_{\mathcal{L}_\mathsf{B}} + \mathsf{S}_{\mathcal{L}_\mathsf{B}}\mathsf{T}_{\mathcal{L}_\mathsf{B}})$
$\rightsquigarrow$ products of block (sub)-diagonal $r_A \times n$ and $r_B \times n$



$\rightsquigarrow O(s^{\omega-1}n)$
But: shuffling by permutations: loose structure $\rightsquigarrow$ only $O(n^2 s^{\omega-2})$

Multiplying two quasiseparable matrices

Quasiseparable $\times$ Quasiseparable

$$A \times B = (L_A + D_A + U_A)(L_B + D_B + U_B).$$

$\times$	Lower	Upper
Lower	$J_n \times \text{Left} \times J_n \times \text{Left}$	$J_n \times \text{Left} \times \text{Left} \times J_n$
Upper	$\text{Left} \times J_n \times J_n \times \text{Left}$	$\text{Left} \times J_n \times \text{Left} \times J_n$

$\rightsquigarrow$ reduce to the complexity of Left-triangular $\times$ Left-triangular

Perspectives

Improve QS $\times$ QS complexity

- r_1-QS $\times$ r_2-QS is $(r_1 + r_2)$-QS
- Classic complexities $(r_1 + r_2)^2 n$

Perspectives

Improve QS $\times$ QS complexity

- r_1-QS $\times$ r_2-QS is $(r_1 + r_2)$-QS
- Classic complexities $(r_1 + r_2)^2 n$
- So far we only get $O((r_1 + r_2)^{\omega-2} n^2)$
- Short way to $O((r_1 + r_2)^{\omega-1} n)$: deal with permutations destroying the stucture

Perspectives

Improve QS $\times$ QS complexity

- r_1-QS $\times$ r_2-QS is $(r_1 + r_2)$-QS
- Classic complexities $(r_1 + r_2)^2 n$
- So far we only get $O((r_1 + r_2)^{\omega-2} n^2)$
- Short way to $O((r_1 + r_2)^{\omega-1} n)$: deal with permutations destroying the stucture

Faster probabilistic elimination

- [**SY14**] propose a $\tilde{O}(n^2 + r^\omega)$ probabilistic algorithm computing the row rank profile
- In [**DPS16**] we extend it to the computation of the Rank Profile Matrix

Perspectives

Improve QS $\times$ QS complexity

- r_1-QS $\times$ r_2-QS is $(r_1 + r_2)$-QS
- Classic complexities $(r_1 + r_2)^2 n$
- So far we only get $O((r_1 + r_2)^{\omega-2} n^2)$
- Short way to $O((r_1 + r_2)^{\omega-1} n)$: deal with permutations destroying the stucture

Faster probabilistic elimination

- [**SY14**] propose a $\tilde{O}(n^2 + r^\omega)$ probabilistic algorithm computing the row rank profile
- In [**DPS16**] we extend it to the computation of the Rank Profile Matrix
- Does it applies to computing
  - the QS orders ?
  - a (compact) Bruhat generator ?

References

A. Bostan, C.-P. Jeannerod, and É. Schost. "Solving structured linear systems with large displacement rank". In: *Theoretical Computer Science* 407.1–3 (Nov. 2008), pp. 155–181. DOI: 10.1016/j.tcs.2008.05.014.

S. Chandrasekaran, M. Gu, and T. Pals. "A Fast ULV Decomposition Solver for Hierarchically Semiseparable Representations". In: *SIAM Journal on Matrix Analysis and Applications* 28.3 (Jan. 2006), pp. 603–622. DOI: 10.1137/S0895479803436652.

J.-G. Dumas, C. Pernet, and Z. Sultan. "Computing the Rank Profile Matrix". In: *Proc. ISSAC'15*. Bath, United Kingdom: ACM, 2015, pp. 149–156. DOI: 10.1145/2755996.2756682.

J.-G. Dumas, C. Pernet, and Z. Sultan. "Simultaneous computation of the row and column rank profiles". In: *Proc. ISSAC'13*. Ed. by M. Kauers. ACM Press, 2013, pp. 181–188. DOI: 10.1145/2465506.2465517.

Y. Eidelman and I. Gohberg. "On a new class of structured matrices". en. In: *Integral Equations and Operator Theory* 34.3 (Sept. 1999), pp. 293–324. DOI: 10.1007/BF01300581.

K. Lessel, M. Hartman, and S. Chandrasekaran. *A fast memory efficient construction algorithm for hierarchically semi-separable representations*. Tech. rep. 2016.

W. Manthey and U. Helmke. "Bruhat canonical form for linear systems". In: *Linear Algebra and its Applications* 425.2–3 (2007). Special Issue in honor of Paul Fuhrmann, pp. 261–282. DOI: 10.1016/j.laa.2007.01.022.

C. Pernet. "Computing with Quasiseparable Matrices". In: *Proc. ISSAC'16*. hal-01264131. 2016.