Block Low-Rank (BLR) approximations to improve multifrontal sparse solvers

Joint work with Patrick Amestoy, Cleve Ashcraft, Olivier Boiteau, Alfredo Buttari and Jean-Yves L’Excellent, PhD started on October 1st, 2010 and financed by EDF.

Clément Weisbecker, ENSEEIHT-IRIT, University of Toulouse, France
### Introduction

Block **Low-Rank approximations** to improve **multifrontal sparse solvers**

#### Multifrontal solver

- direct solver for large linear systems
- objective: $A = LU$

#### Low-rank approximations

- compression and flop reduction
- accuracy controlled by a numerical parameter

⇒ Combine these two notions to improve multifrontal solvers (in the context of **MUMPS**)

---

The multifrontal method
The multifrontal method [Duff & Reid ’83]

Nested dissection

Elimination tree
The multifrontal method [Duff & Reid ‘83]

Nested dissection

Fully-summed variables (FS) = separator

Elimination tree
The multifrontal method [Duff & Reid '83]

Nested dissection

Elimination tree

Non fully-summed variables (NFS) = border
The multifrontal method [Duff & Reid ‘83]

Nested dissection

Non fully-summed variables (NFS) = border

⇒ stack of CBs
Low-rank approximations
Consider a block $B$ of size $m \times n$ and $k_\epsilon$ its approximated numerical rank at accuracy $\epsilon$. $B$ is said to be low-rank if it can be written as

$$B = X \cdot Y + E \text{ with } \|E\|_2 \leq \epsilon \text{ and } k_\epsilon(m + n) < mn$$

If $B$ is low-rank, storing it as $X, Y$ saves storage and allows faster operations. $X, Y$ can be computed using rank-revealing $QR$, SVD…

Low-rank product: $X_1(Y^T_1 X_2)Y^T_2$
Can we exploit low-rankness in multifrontal methods?

- Fronts are not low-rank but in many applications they exhibit some low-rank blocks.

**Idea:** find and compress low rank blocks within frontal matrices.  
**Problem:** how to identify low-rank blocks?

⇒ Define a *clustering* $\mathcal{C}$ to obtain low-rank blocks $A_b$  
($b = \sigma \times \tau \subset I \times I$).
How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:
How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:

\[
\text{diam}(σ) \quad \text{dist}(σ, τ) \quad \text{diam}(τ)
\]

\[
\begin{array}{c}
\sigma \\
\end{array} \quad \begin{array}{c}
\tau \\
\end{array}
\]
How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:
How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:

\[ \text{diam}(\sigma) \quad \text{dist}(\sigma, \tau) \quad \text{diam}(\tau) \]

\[ \sigma \quad \tau \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]

\[ \begin{array}{c}
\sigma \\
\tau
\end{array} \]
Admissibility condition

How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:

\[
\text{diam}(\sigma) \quad \text{dist}(\sigma, \tau) \quad \text{diam}(\tau)
\]
Admissibility condition

How to find a good clustering?

⇒ Admissibility condition [Börm, Grasedyck, Hackbusch] expresses correlation between distance and rank:

\[ \text{diam}(\sigma) \quad \text{dist}(\sigma, \tau) \quad \text{diam}(\tau) \]

\[ \sigma \]

\[ \tau \]

\[ \text{rank of } \sigma = 80 \]

\[ \text{rank of } \tau = 128 \]

\[ \text{distance between } \tau \text{ and } \sigma \]

\[ \text{rank of } \Lambda_{tr} \]
• There are different low-rank representations (heuristics to exploit low-rank blocks): $\mathcal{H}, \mathcal{H}^2$ [Bebendof, Börm, Hackbush, Grasedyck,...], Hierarchical/Sequential Semiseparable (HSS/SSS) [Chandrasekaran, Dewilde, Gu, Li, Xia,...], BLR [Amestoy et al.], etc.

• Some representations are simpler and apply to broader classes of problems but provide less gain in memory/operations, while some others are more complex but allow for further gains in complexity.

• We focus on Block Low-Rank (BLR).
HSS vs BLR

\[
HSS = \begin{bmatrix}
D_1 & X_1B_1Y_2^T \\
X_2B_2Y_1^T & D_2 \\
X_4R_4B_6W_1^TY_1^T & X_4R_4B_6W_2^TY_2^T \\
X_5R_5B_6W_1^TY_1^T & X_5R_5B_6W_2^TY_2^T \\
\end{bmatrix}
\begin{bmatrix}
X_1R_1B_3W_4^TY_4^T \\
X_2R_2B_3W_4^TY_4^T \\
X_4B_4Y_5^T \\
\end{bmatrix}
\]

\[
BLR = \begin{bmatrix}
D_1 & X_{12}Y_{12}^T & X_{13}Y_{13}^T & X_{14}Y_{14}^T \\
X_{21}Y_{21}^T & D_2 & X_{23}Y_{23}^T & X_{24}Y_{24}^T \\
X_{31}Y_{31}^T & X_{32}Y_{32}^T & D_3 & X_{34}Y_{34}^T \\
X_{41}Y_{41}^T & X_{42}Y_{42}^T & X_{43}Y_{43}^T & D_4 \\
\end{bmatrix}
\]

⇒ particular case of $\mathcal{H}$-matrices
⇒ no tree
⇒ natural matrix structure
Comparative study: compression rates

Compression rates of the frontal matrix at the root of a multifrontal tree, on two 3D stencils (discretization of a 128 x 128 x 128 cube)

Laplacian 27-pts Geoazur stencil

<table>
<thead>
<tr>
<th>low-rank threshold</th>
<th>0.05</th>
<th>0.1</th>
<th>0.15</th>
<th>0.2</th>
<th>0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HSS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

fraction of dense storage

27-pts Geoazur stencil

<table>
<thead>
<tr>
<th>low-rank threshold</th>
<th>0.05</th>
<th>0.1</th>
<th>0.15</th>
<th>0.2</th>
<th>0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HSS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

fraction of dense storage
Comparative study: compression cost

Compression cost of the frontal matrix at the root of a multifrontal tree, on two 3D stencils (discretization of a 128 x 128 x 128 cube)

Laplacian 27-pts Geoazur stencil

\[ \log_{10} \text{(number of operations for compression)} \]

<table>
<thead>
<tr>
<th>Low-rank threshold</th>
<th>BLR</th>
<th>HSS</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>(10^{-14})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-12})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-10})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-8})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-6})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-4})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10^{-2})</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \log_{10} \text{(number of operations for compression)} \]

BLR
HSS
H

Low-rank threshold

9
9.5
10
10.5
11
11.5
12
12.5
13

\[ \text{log}_{10} \text{(number of operations for compression)} \]

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

10
10
10
10
10
10
10
10

Adaptation to a multifrontal solver

Adaptation to a multifrontal solver

no relative order $\Rightarrow$ efficient clustering

Adaptation to a multifrontal solver

- no relative order
- efficient compr. and decompr.
- efficient clustering
- distribution
- assembly

\[
\begin{align*}
L_{1,1}^i &; U_{1,1}^i \\
L_{2,2}^i &; U_{2,2}^i \\
& \quad \vdots \\
L_{d_i,d_i}^i &; U_{d_i,d_i}^i \\
\end{align*}
\]

\[
\text{CB}
\]
Adaptation to a multifrontal solver

- No relative order
- Efficient compression and decompression
- Flat non-hierarchical structure
- Efficient clustering
- Distribution
- Assembly
- Pivoting

\[ L^i_{1,1} U^i_{1,1} \]
\[ L^i_{2,2} U^i_{2,2} \]
\[ L^i_{d_i,d_i} U^i_{d_i,d_i} \]

CB
Clustering variables
Admissible clustering

**Constraint**: the admissibility condition should be satisfied

- **large diameters**
  - fraction of memory used 83%

- **small diameters**
  - fraction of memory used 57%
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool

1. The separator
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool

1. The separator
2. The halo
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool

1. The separator
2. The halo
3. Extraction of the halo
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool

1. The separator
2. The halo
3. Extraction of the halo
4. Partition of the halo
Halo algorithm for the clustering of a separator

- Designed to catch the geometry of the problem
- Computed with the graph instead of the mesh
- Coupled with a third party partitioning tool

1. The separator
2. The halo
3. Extraction of the halo
4. Partition of the halo
5. Partition of the separator (block size is fixed)
Clustering of the variables of a front

$$\Rightarrow \text{front} = \text{separator} + \text{border}$$
Clustering of the variables of a front

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1. separator : halo
Clustering of the variables of a front

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1. separator : halo
2. border ? 2 choices :
Clustering of the variables of a front

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1- separator : halo

2- border ? 2 choices :

EXPLICIT

```
/\ S
|   |
|_S|
```

Separator: halo
Clustering of the variables of a front

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1- separator: halo

2- border? 2 choices:

EXPLICIT

\[ S \]
Clustering of the variables of a front

\[ \text{front} = \text{separator} + \text{border} \]

1- separator : halo
2- border ? 2 choices :

EXPLICIT

INHERITED (top down)
Clustering of the variables of a front

---

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1- separator : halo

2- border ? 2 choices :

- EXPLICIT

- INHERITED (top down)
Clustering of the variables of a front

\[ \Rightarrow \text{front} = \text{separator} + \text{border} \]

1. separator: halo
2. border? 2 choices:

**EXPLICIT**

**INHERITED (top down)**
Clustering of the variables of a front

front = separator + border

1- separator: halo
2- border? 2 choices:

EXPLICIT

INHERITED (top down)
Front structure with inherited clustering

- optimal × optimal = optimal block
Front structure with inherited clustering

- optimal \times optimal = optimal block
- small \times optimal = large enough block
Front structure with inherited clustering

- optimal × optimal = optimal block
- small × optimal = large enough block
- small × small = too small block
Front structure with inherited clustering

- optimal $\times$ optimal = optimal block
- small $\times$ optimal = large enough block
- small $\times$ small = too small block

$\Rightarrow$ reclustering strategies
Block Low-Rank multifrontal method
### Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

**Standard FR algorithm:**

- F Factor
- S Solve
- U Update
# Factorization Algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = L U^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = X Y^T$</td>
<td>$n^2 k$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X (Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$k n^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1 (Y_1^T X_2) Y^T_2$</td>
<td>$2n^3$</td>
<td>$2k n^2$</td>
</tr>
</tbody>
</table>

**Standard FR Algorithm:**

- **F Factor**
- **S Solve**
- **U Update**
### Factorization Algorithms

<table>
<thead>
<tr>
<th>Task</th>
<th>Operation Type</th>
<th>Dense</th>
<th>Low-Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>( B = LU^T )</td>
<td>( \frac{2}{3}n^3 )</td>
<td>( \frac{2}{3}n^3 )</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>( C = XY^T )</td>
<td>( kn^2 )</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>( D = X(Y^TL^{-1}) )</td>
<td>( n^3 )</td>
<td>( kn^2 )</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>( D = D - X_1(Y_1^TX_2)Y_2^T )</td>
<td>( 2n^3 )</td>
<td>( 2kn^2 )</td>
</tr>
</tbody>
</table>

**Standard FR Algorithm:**

- **F** Factor
- **S** Solve
- **U** Update
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = L U^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = X Y^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X (Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1 (Y_1^T X_2) Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

### Standard FR algorithm:

- **F Factor**
- **S Solve**
- **U Update**
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^TL^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^TX_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

#### BLR FSCU algorithm:

- F Factor
- S Solve
- U Update
### Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = L U^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = X Y^T$</td>
<td>$k n^2$</td>
<td>$-$</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X (Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$k n^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2k n^2$</td>
</tr>
</tbody>
</table>

**BLR FSCU algorithm:**

- **F Factor**
- **S Solve**
- **U Update**
Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

BLR FScU algorithm:

F Factor

S Solve

U Update
### Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = L U^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = X Y^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X (Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1 (Y_1^T X_2) Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

**BLR FSCU algorithm:**

- **F** Factor
- **S** Solve
- **C** Compress
- **U** Update
## Factorization Algorithms

<table>
<thead>
<tr>
<th>Task</th>
<th>Operation Type</th>
<th>Dense</th>
<th>Low-Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

### BLR FSCU Algorithm:

- **F** Factor
- **S** Solve
- **C** Compress
- **U** Update
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^T L^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^T X_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

F Factor
S Solve
C Compress
U Update

**FCSU version**
more efficient
less stability
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>$B = LU^T$</td>
<td>$(2/3)n^3$</td>
<td>$(2/3)n^3$</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>$C = XY^T$</td>
<td>$kn^2$</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>$D = X(Y^TL^{-1})$</td>
<td>$n^3$</td>
<td>$kn^2$</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>$D = D - X_1(Y_1^TX_2)Y_2^T$</td>
<td>$2n^3$</td>
<td>$2kn^2$</td>
</tr>
</tbody>
</table>

F Factor

S Solve

C Compress

U Update

**FSUC version**

no flop reduction

more stability
## Factorization algorithms

<table>
<thead>
<tr>
<th>task</th>
<th>operation type</th>
<th>dense</th>
<th>low-rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor (F)</td>
<td>( B = LU^T )</td>
<td>((2/3)n^3)</td>
<td>((2/3)n^3)</td>
</tr>
<tr>
<td>Compress (C)</td>
<td>( C = XY^T )</td>
<td>(kn^2)</td>
<td>—</td>
</tr>
<tr>
<td>Solve (S)</td>
<td>( D = X(Y^T L^{-1}) )</td>
<td>(n^3)</td>
<td>(kn^2)</td>
</tr>
<tr>
<td>CB update (U)</td>
<td>( D = D - X_1(Y_1^T X_2)Y_2^T )</td>
<td>(2n^3)</td>
<td>(2kn^2)</td>
</tr>
</tbody>
</table>

F Factor  
S Solve  
C Compress  
U Update  

High algorithmic flexibility
Experiments
<table>
<thead>
<tr>
<th>Name</th>
<th>Prop.</th>
<th>Arith.</th>
<th>N</th>
<th>NZ</th>
<th>mem. LU</th>
<th>flops LU</th>
<th>CSR</th>
<th>appli.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curl 5000²</td>
<td>2D/sym. D</td>
<td>50</td>
<td>0.2</td>
<td>29</td>
<td>5</td>
<td>2E-15</td>
<td>▼</td>
<td></td>
</tr>
<tr>
<td>Geoazur 128³</td>
<td>3D/unsym. Z</td>
<td>2</td>
<td>55</td>
<td>54</td>
<td>60</td>
<td>3E-4</td>
<td>wave prop.</td>
<td></td>
</tr>
<tr>
<td>EDF_A_MECA_R12</td>
<td>2D/sym. D</td>
<td>134</td>
<td>1</td>
<td>151</td>
<td>200</td>
<td>4E-15</td>
<td>mechanics</td>
<td></td>
</tr>
<tr>
<td>EDF_D_THER_R7</td>
<td>3D/sym. D</td>
<td>8</td>
<td>118</td>
<td>229</td>
<td>100</td>
<td>8E-15</td>
<td>thermics</td>
<td></td>
</tr>
</tbody>
</table>

- CSR = Componentwise Scaled Residual
- Code_Aster tpl1101{a,d} test cases (refined)
- large matrices
- applicative problems
memory:

- $|L|$ = fraction of FR factors storage obtained with BLR (%)
- $|CB|$ = fraction of FR maximum size of CB stack obtained with BLR (%)

flops:

- fraction of FR operations needed for the BLR factorization (in percent or absolute data, including the compression cost)
Clustering strategy

<table>
<thead>
<tr>
<th>clustering</th>
<th>memory</th>
<th></th>
<th>flops</th>
<th></th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$</td>
<td>L</td>
<td>\text{inh}\text{ exp}$</td>
<td>$</td>
</tr>
<tr>
<td>Cur15000</td>
<td>63.7%</td>
<td>62.4%</td>
<td>7.0%</td>
<td>5.5%</td>
<td>10.9%</td>
</tr>
<tr>
<td>Geoazur128</td>
<td>79.0%</td>
<td>77.0%</td>
<td>47.0%</td>
<td>45.0%</td>
<td>60.8%</td>
</tr>
<tr>
<td>TH_RAFF7</td>
<td>34.1%</td>
<td>30.7%</td>
<td>17.5%</td>
<td>16.2%</td>
<td>7.2%</td>
</tr>
<tr>
<td>ME_RAFF12</td>
<td>52.9%</td>
<td>51.1%</td>
<td>4.8%</td>
<td>4.1%</td>
<td>6.1%</td>
</tr>
</tbody>
</table>

- “inherited” version is more than 2 times faster
- same results on $L_{11}$
- comparable results on $L_{21}$
- a little less good on CBs

⇒ **inherited** clustering used for all the experiments
Results with different orderings on Geoazur128 problem.

| ordering | mry | flops   | peak  | |L|  | |CB|  | flops |
|----------|-----|---------|-------|-----|-----|-------|-----|-----|
| AMD      | 109GB | 3.9E + 14 | 92GB  | 73.5% | 40.0% | 59.4% |
| AMF      | 72GB  | 1.8E + 14 | 45GB  | 69.9% | 53.5% | 48.3% |
| PORD     | 55GB  | 1.0E + 14 | 28GB  | 70.4% | 49.0% | 48.9% |
| METIS    | 46GB  | 6.2E + 13 | 20GB  | 78.7% | 46.4% | 62.6% |
| SCOTCH   | 49GB  | 6.6E + 13 | 21GB  | 79.4% | 48.1% | 63.4% |
Global ordering of the matrix: general results

Results with different orderings on Geoazur128 problem.

<table>
<thead>
<tr>
<th>ordering</th>
<th>mry</th>
<th>flops</th>
<th>peak</th>
<th>FR</th>
<th>mry</th>
<th>flops</th>
<th>peak</th>
<th>LR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMD</td>
<td>109GB</td>
<td>$3.9E + 14$</td>
<td>92GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMF</td>
<td>72GB</td>
<td>$1.8E + 14$</td>
<td>45GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PORD</td>
<td>55GB</td>
<td>$1.0E + 14$</td>
<td>28GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>METIS</td>
<td>46GB</td>
<td>$6.2E + 13$</td>
<td>20GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCOTCH</td>
<td>49GB</td>
<td>$6.6E + 13$</td>
<td>21GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Algebraic approach
Global ordering of the matrix: SCOTCH tree

[visualization tool developed by M. Bremond]
Influence of the size of the problem

⇒ different refinements of problem EDF_A_MECA done with Homard

<table>
<thead>
<tr>
<th>Refinement</th>
<th>N</th>
<th>memory</th>
<th>flops</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
</tr>
<tr>
<td>R8</td>
<td>2,101,258</td>
<td>71%</td>
<td>43%</td>
</tr>
<tr>
<td>R11</td>
<td>33,570,826</td>
<td>59%</td>
<td>29%</td>
</tr>
<tr>
<td>R12</td>
<td>134,250,506</td>
<td>53%</td>
<td>24%</td>
</tr>
</tbody>
</table>

- the larger the problem, the more efficient the method
- target = challenging problems
Scalability with respect to the size of the problem

⇒ Laplacian problem, $\varepsilon = 10^{-14}$

$\Rightarrow O(N^{4/3})$ complexity (≡ HSS)
Global results: 2D problems

<table>
<thead>
<tr>
<th>Name</th>
<th>N</th>
<th>mem LU</th>
<th>flops LU</th>
<th>CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDF_A_MECA_R12</td>
<td>134E+6</td>
<td>151 GB</td>
<td>2E+14</td>
<td>4E-15</td>
</tr>
<tr>
<td>Curl-Curl 5000²</td>
<td>50E+6</td>
<td>29 GB</td>
<td>5E+12</td>
<td>2E-15</td>
</tr>
</tbody>
</table>
Global results: 3D problems

<table>
<thead>
<tr>
<th>Name</th>
<th>N</th>
<th>mem LU</th>
<th>flops LU</th>
<th>CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDF_D_THER_R7</td>
<td>8E+6</td>
<td>229 GB</td>
<td>1E+14</td>
<td>8E-15</td>
</tr>
<tr>
<td>Geoazar 128³</td>
<td>2E+6</td>
<td>54 GB</td>
<td>6E+13</td>
<td>3E-4</td>
</tr>
</tbody>
</table>
Application to geophysics (1)

- Helmholtz equation for seismic modeling (SEISCOPE project)
- EAGE overthrust ground model
- Single precision computations

<table>
<thead>
<tr>
<th>fqc</th>
<th>Flops LU</th>
<th>Mem LU</th>
<th>Peak memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Hz</td>
<td>8.957E+11</td>
<td>3 GB</td>
<td>4 GB</td>
</tr>
<tr>
<td>4 Hz</td>
<td>1.639E+13</td>
<td>22 GB</td>
<td>25 GB</td>
</tr>
<tr>
<td>8 Hz</td>
<td>5.769E+14</td>
<td>247 GB</td>
<td>283 GB</td>
</tr>
</tbody>
</table>

Application to geophysics (2)

<table>
<thead>
<tr>
<th>$\varepsilon$</th>
<th>freqy</th>
<th>facto</th>
<th>memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(10^{-5})$</td>
<td>2 Hz</td>
<td>41.8%</td>
<td>61.8%</td>
</tr>
<tr>
<td></td>
<td>4 Hz</td>
<td>27.4%</td>
<td>50.0%</td>
</tr>
<tr>
<td></td>
<td>8 Hz</td>
<td>21.8%</td>
<td>41.6%</td>
</tr>
<tr>
<td>$(10^{-4})$</td>
<td>2 Hz</td>
<td>32.9%</td>
<td>53.4%</td>
</tr>
<tr>
<td></td>
<td>4 Hz</td>
<td>20.0%</td>
<td>42.2%</td>
</tr>
<tr>
<td></td>
<td>8 Hz</td>
<td>15.2%</td>
<td>28.9%</td>
</tr>
<tr>
<td>$(10^{-3})$</td>
<td>2 Hz</td>
<td>24.6%</td>
<td>44.7%</td>
</tr>
<tr>
<td></td>
<td>4 Hz</td>
<td>13.8%</td>
<td>34.5%</td>
</tr>
<tr>
<td></td>
<td>8 Hz</td>
<td>9.8%</td>
<td>21.3%</td>
</tr>
</tbody>
</table>

Preconditioning with BLR: set of problems

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>NZ</th>
<th>Cond.</th>
<th>application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Piston</td>
<td>1.3E+6</td>
<td>54.7E+6</td>
<td>5.1E+5</td>
<td>external pressure force on the top</td>
</tr>
<tr>
<td>perf001d</td>
<td>2.0E+6</td>
<td>75.8E+6</td>
<td>1.5E+11</td>
<td>“cavity” hook subjected to internal pressure force (challenging for EDF)</td>
</tr>
</tbody>
</table>

- CG preconditioned with MUMPS single precision with BLR
- preliminary study
### Preconditioning with BLR: results

**perf001d, FR SP = 50 iterations**

| $\varepsilon$   | #it | $|L|$ | $|CB|$ | flops | #it | $|L|$ | $|CB|$ | flops |
|-----------------|-----|------|-------|-------|-----|------|-------|-------|
| $10^{-10}$      | –   | –    | –     | –     | 2   | 64.2%| 18.3% | 31.0% |
| $10^{-9}$       | –   | –    | –     | –     | 2   | 62.2%| 16.1% | 28.8% |
| $10^{-8}$       | 67  | 59.4%| 18.9% | 26.7% | 3   | 58.7%| 13.7% | 25.5% |
| $10^{-7}$       | 68  | 56.9%| 16.9% | 24.3% | 4   | 56.4%| 11.4% | 23.4% |
| $10^{-6}$       | 66  | 52.4%| 15.2% | 20.2% | 8   | 51.9%| 9.6%  | 19.7% |
| $10^{-5}$       | 67  | 49.1%| 14.1% | 17.1% | 19  | 48.6%| 8.6%  | 17.0% |
| $10^{-4}$       | 81  | 45.1%| 13.5% | 14.1% | 68  | 44.4%| 8.0%  | 14.1% |

Is it the right definition of optimality?
Preconditioning with BLR: results

**perf001d, FR SP = 50 iterations**

| $\varepsilon$ | #it | $|L|$ | $|CB|$ | flops  | $\varepsilon$ | $|L|$ | $|CB|$ | flops  |
|---------------|-----|------|------|--------|---------------|------|------|--------|
| $10^{-10}$    | –   | –    | –    | –      | $2$           | 64.2%| 18.3%| 31.0%  |
| $10^{-9}$     | –   | –    | –    | –      | $2$           | 62.2%| 16.1%| 28.8%  |
| $10^{-8}$     | 67  | 59.4%| 18.9%| 26.7%  | $3$           | 58.7%| 13.7%| 25.5%  |
| $10^{-7}$     | 68  | 56.9%| 16.9%| 24.3%  | $4$           | 56.4%| 11.4%| 23.4%  |
| $10^{-6}$     | 66  | 52.4%| 15.2%| 20.2%  | $8$           | 51.9%| 9.6% | 19.7%  |
| $10^{-5}$     | 67  | 49.1%| 14.1%| 17.1%  | $19$          | 48.6%| 8.6% | 17.0%  |
| $10^{-4}$     | 81  | 45.1%| 13.5%| 14.1%  | $68$          | 44.4%| 8.0% | 14.1%  |
Preconditioning with BLR: results

perf001d, FR SP = 50 iterations

| $\varepsilon$ | #it | |L| |CB| |BLR SP| |#it | |L| |CB| |BLR DP| |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $10^{-10}$ | – | – | – | – | 2 | 64.2% | 18.3% | 31.0% |
| $10^{-9}$  | – | – | – | – | 2 | 62.2% | 16.1% | 28.8% |
| $10^{-8}$  | 67 | 59.4% | 18.9% | 26.7% | 3 | 58.7% | 13.7% | 25.5% |
| $10^{-7}$  | 68 | 56.9% | 16.9% | 24.3% | 4 | 56.4% | 11.4% | 23.4% |
| $10^{-6}$  | 66 | 52.4% | 15.2% | 20.2% | 8 | 51.9% | 9.6% | 19.7% |
| $10^{-5}$  | 67 | 49.1% | 14.1% | 17.1% | 19 | 48.6% | 8.6% | 17.0% |
| $10^{-4}$  | 81 | 45.1% | 13.5% | 14.1% | 68 | 44.4% | 8.0% | 14.1% |
Preconditioning with BLR: results

**perf001d, FR SP = 50 iterations**

| ε      | #it | |L| | |CB| | flops | #it | |L| | |CB| | flops |
|--------|-----|---|---|---|----|---|-----|-----|---|---|---|---|-----|
| 10^{-10} | -   | - | - | - | -  | 2  | 64.2% | 18.3% | 31.0% |
| 10^{-9}  | -   | - | - | - | -  | 2  | 62.2% | 16.1% | 28.8% |
| 10^{-8}  | 67  | 59.4% | 18.9% | 26.7% | 3  | 58.7% | 13.7% | 25.5% |
| 10^{-7}  | 68  | 56.9% | 16.9% | 24.3% | 4  | 56.4% | 11.4% | 23.4% |
| 10^{-6}  | 66  | 52.4% | 15.2% | 20.2% | 8  | 51.9% | 9.6% | 19.7% |
| 10^{-5}  | 67  | 49.1% | 14.1% | 17.1% | 19 | 48.6% | 8.6% | 17.0% |
| 10^{-4}  | 81  | 45.1% | 13.5% | 14.1% | 68 | 44.4% | 8.0% | 14.1% |

- is it the right definition of *optimality*?
Preconditioning with BLR: \texttt{perf001d} timings

![Graph showing the relationship between QR dropping parameter and time in seconds for total execution time, factorization time, and solves time (GCPC).]

- Total execution time: FR = 569s
- Factorization time: FR = 452s
- Solves time (GCPC): FR = 117s
Preconditioning with BLR: Piston timings

![Graph showing Piston timings with time in seconds on the x-axis and execution time in seconds on the y-axis, with different parameters and execution times indicated.]

<table>
<thead>
<tr>
<th>FR</th>
<th>10^{-8}</th>
<th>10^{-7}</th>
<th>10^{-6}</th>
<th>10^{-5}</th>
<th>10^{-4}</th>
<th>10^{-3}</th>
<th>10^{-2}</th>
</tr>
</thead>
<tbody>
<tr>
<td># iterations</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>115</td>
</tr>
</tbody>
</table>
Conclusion & perspectives

- efficient method on various applicative problems
- considerable memory reduction & substantial decrease in computations $\Rightarrow O(N^{4/3})$ complexity on a Laplacian, comparable to HSS
- can be used as a preconditioner or as a direct solver
- good potential for parallelism
- code is stable on tested problems

- MPI
- study on larger and more difficult problems
- error propagation study $\Rightarrow$ absolute or relative dropping parameter? relative to what?
  (work with S. Gratton, M. Ngom and D. Titley-Peloquin started)
• O. Boiteau, B. Quinnez and N. Tardieu (EDF R&D)
• S. Operto, R. Brossier and J. Virieux (SEISCOPE Project) for their contribution to the geophysics study
• the Toulouse Computing Center (CICT) and N. Renon
• S. Li, A. Napov and F.-H. Rouet (LBNL Berkeley)

Details on this work can be found in:

Thank you!
Any questions?