

Solveurs multifrontaux exploitant des blocs de rang faible: complexité, performance et parallélisme

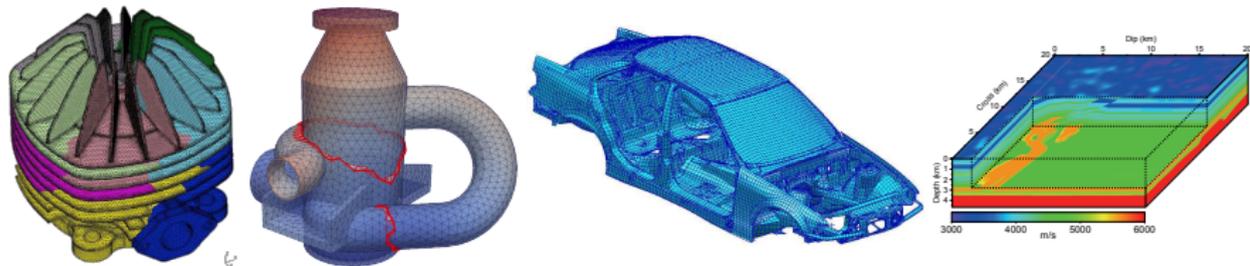
Thèse préparée à l'IRIT sous la direction de Patrick Amestoy et Alfredo Buttari

Théo Mary

University of Manchester, School of Mathematics

Congrès de la SIF, Bordeaux, 6-7 Février 2019

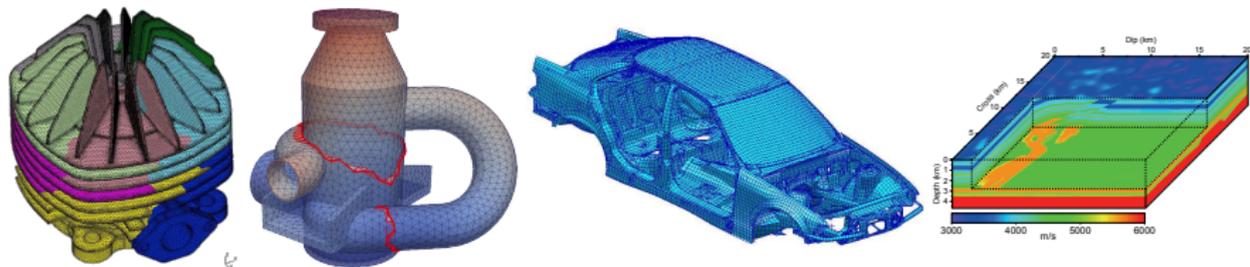




Problème physique discrétisé \Rightarrow **système d'équations linéaires**

Exemple à 2 équations:

$$\begin{cases} ax + by = c \\ a'x + b'y = c' \end{cases}$$



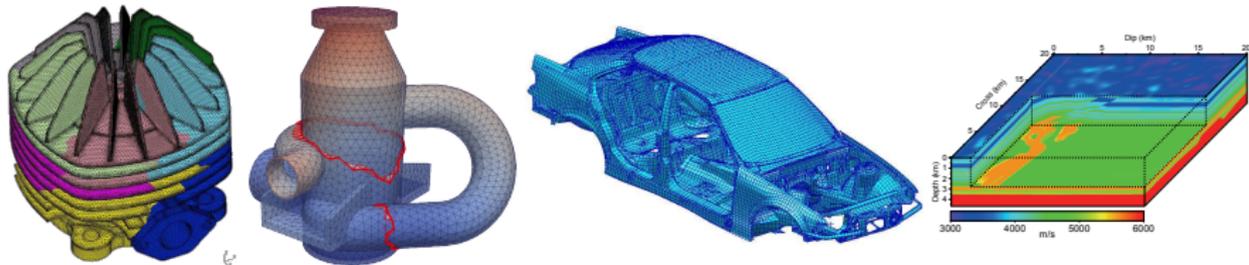
Problème physique discrétisé \Rightarrow **système d'équations linéaires**

Exemple à 2 équations:

Expression matricielle:

$$\begin{cases} ax + by = c \\ a'x + b'y = c' \end{cases} \Rightarrow \begin{pmatrix} a & b \\ a' & b' \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} c \\ c' \end{pmatrix}$$

A **x** = **b**



Problème physique discrétisé \Rightarrow **système d'équations linéaires**

Exemple à 2 équations:

Expression matricielle:

$$\begin{cases} ax + by = c \\ a'x + b'y = c' \end{cases} \Rightarrow \begin{pmatrix} a & b \\ a' & b' \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} c \\ c' \end{pmatrix}$$

A **x** = **b**

Milliers, voire **millions d'équations**

\Rightarrow Besoin de **supercalculateurs**



Deux classes de méthodes pour résoudre $Ax = b$ (A d'ordre n)

Méthodes itératives

Construction d'une suite x_k convergeant vers x

- 😊 Faible coût calculatoire: $\mathcal{O}(n)$ opérations/itération et stockage
- ☹ Leur convergence dépend des propriétés de la matrice
⇒ **limitées en gamme de problèmes**

Méthodes directes

Factorisation de $A = LU$ et résolution triangulaire $LUx = b$

- 😊 Grande fiabilité numérique
- ☹ Coût calculatoire élevé: $\mathcal{O}(n^2)$ opérations, $\mathcal{O}(n^{4/3})$ stockage
⇒ **limitées en taille de problèmes**

Deux classes de méthodes pour résoudre $Ax = b$ (A d'ordre n)

Méthodes itératives

Construction d'une suite x_k convergeant vers x

- ☺ Faible coût calculatoire: $\mathcal{O}(n)$ opérations/itération et stockage
- ☹ Leur convergence dépend des propriétés de la matrice
⇒ **limitées en gamme de problèmes**

Méthodes directes

Factorisation de $A = LU$ et résolution triangulaire $LUx = b$

- ☺ Grande fiabilité numérique
- ☹ Coût calculatoire élevé: $\mathcal{O}(n^2)$ opérations, $\mathcal{O}(n^{4/3})$ stockage
⇒ **limitées en taille de problèmes**

Objectif de ma thèse: concevoir des méthodes capables de résoudre une **large gamme de problèmes de grande taille**

Approximations de rang faible

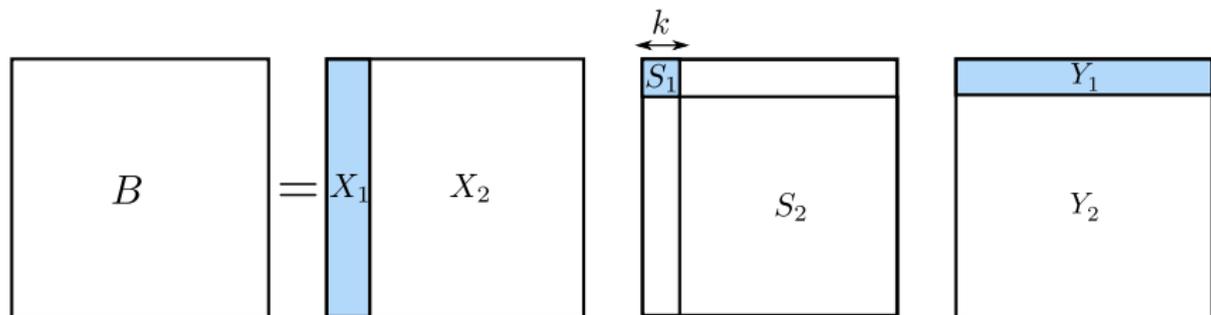
Soit une matrice B de dimensions $b \times b$ et calculons sa décomposition en valeurs singulières $B = XSY$:

$$B = XSY$$

The diagram illustrates the equation $B = XSY$ using four square boxes. The first box on the left contains the letter B . To its right is an equals sign. The second box contains the letter X . To its right is the letter S . To its right is the letter Y . All boxes are empty except for the letters they contain.

Approximations de rang faible

Soit une matrice B de dimensions $b \times b$ et calculons sa décomposition en valeurs singulières $B = XSX^T$:



Soit k_ϵ le nombre de valeur singulières supérieures à un seuil ϵ

Approximations de rang faible

Soit une matrice B de dimensions $b \times b$ et calculons sa décomposition en valeurs singulières $B = XS_1Y$:



Soit k_ε le nombre de valeur singulières supérieures à un seuil ε

$\tilde{B} = X_1 S_1 Y_1$ est une **approximation de rang faible** de B telle que

$$\|B - \tilde{B}\|_2 = \sigma_{k_\varepsilon+1} \leq \varepsilon$$

Pour $k_\varepsilon \ll b$, coût calculatoire fortement réduit en utilisant \tilde{B} à la place de B , tout en contrôlant la perte de précision avec ε

Approximations de rang faible

Soit une matrice B de dimensions $b \times b$ et calculons sa décomposition en valeurs singulières $B = XSY$:



Soit k_ε le nombre de valeur singulières supérieures à un seuil ε

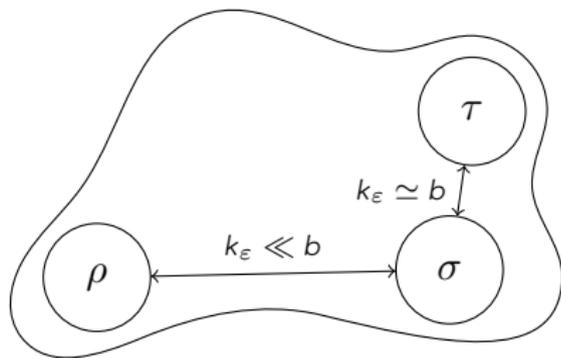
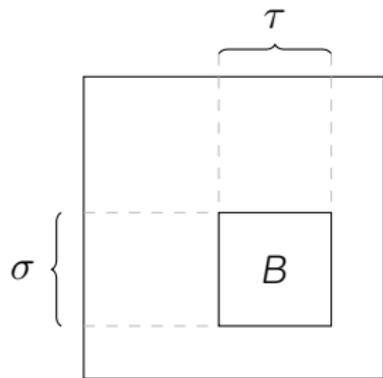
$\tilde{B} = X_1 S_1 Y_1$ est une **approximation de rang faible** de B telle que

$$\|B - \tilde{B}\|_2 = \sigma_{k_\varepsilon+1} \leq \varepsilon$$

Pour $k_\varepsilon \ll b$, coût calculatoire fortement réduit en utilisant \tilde{B} à la place de B , tout en contrôlant la perte de précision avec ε

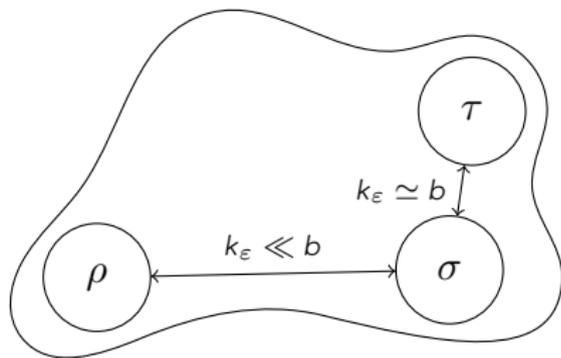
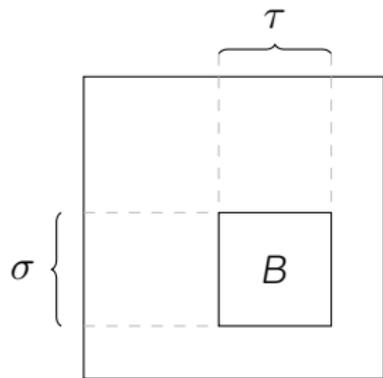
Comment utiliser ces approximations de rang faible pour réduire le coût de résolution de $Ax = b$?

Dans plusieurs applications la matrice A possède des **blocs de rang faible**



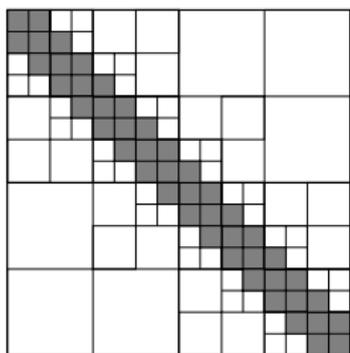
Un bloc B représente l'**interaction** entre deux sous-domaines.
Sous-domaines distants \Rightarrow bloc de **rang faible**

Dans plusieurs applications la matrice A possède des **blocs de rang faible**



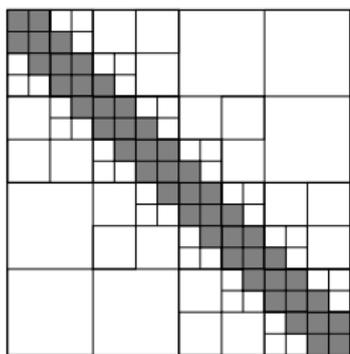
Un bloc B représente l'**interaction** entre deux sous-domaines.
Sous-domaines distants \Rightarrow bloc de rang faible

Comment choisir le découpage de la matrice en blocs?



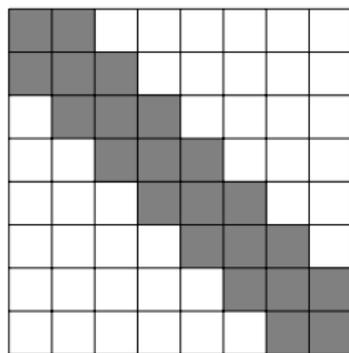
Matrice \mathcal{H}

- Coût calculatoire réduit
 $\mathcal{O}(n^2) \rightarrow \mathcal{O}(n)$
- Structure hiérarchique très complexe, **peu adéquat pour le calcul parallèle**



Matrice \mathcal{H}

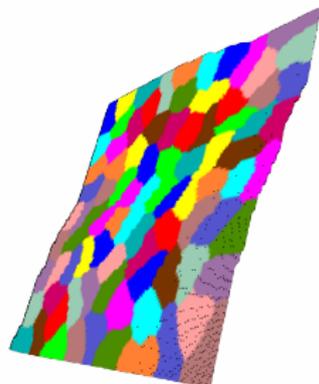
- Coût calculatoire réduit
 $\mathcal{O}(n^2) \rightarrow \mathcal{O}(n)$
- Structure hiérarchique très complexe, **peu adéquat pour le calcul parallèle**



Matrice Block Low-Rank (BLR)

- Structure non hiérarchique, **idéal pour la calcul parallèle**
 - Mais quel coût calculatoire?
 $\mathcal{O}(n^2) \rightarrow ?$
- ⇒ Longtemps rejeté par la communauté qui conjecturait un coût en $\mathcal{O}(n^2)$

- La théorie des matrices \mathcal{H} n'est pas adaptée et donne une complexité $\mathcal{O}(n^2)$ pour les matrices BLR
- Nous avons étendu cette théorie en prouvant que le **nombre de blocs de rang plein par ligne/colonne est constant**



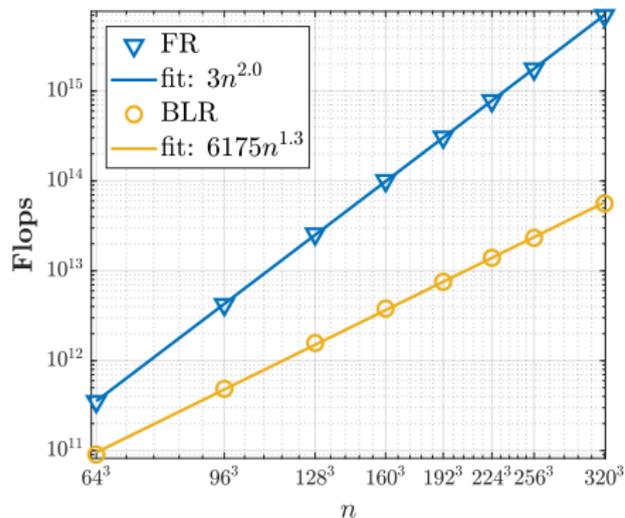
Complexité asymptotique de la factorisation BLR:

- Opérations: $\mathcal{O}(n^2) \rightarrow \mathcal{O}(n^{4/3})$
 - Stockage: $\mathcal{O}(n^{4/3}) \rightarrow \mathcal{O}(n \log n)$
- ⇒ le BLR est donc un **compromis entre complexité et parallélisme**

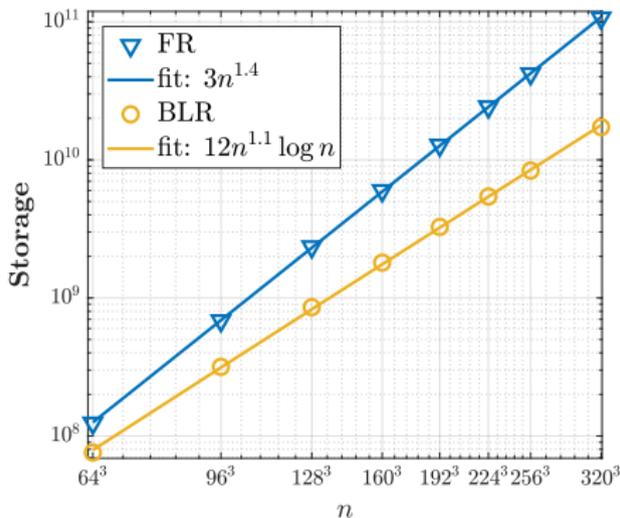


P. Amestoy, A. Buttari, J.-Y. L'Excellent, and T. Mary. *On the Complexity of the Block Low-Rank Multifrontal Factorization*. SIAM J. Sci. Comput. (2017).

Opérations



Stockage



Matrices issues d'un problème de Poisson, seuil $\varepsilon = 10^{-10}$

Complexités mesurées expérimentalement **confirment la théorie**:

- Opérations: $O(n^{4/3}) \rightarrow O(n^{1.3})$
- Stockage: $O(n \log n) \rightarrow O(n^{1.1} \log n)$

Exemple de performances sur une matrice d'ordre **21 millions**

	opérations ($\times 10^{12}$)	temps (24 cœurs)
Référence	3119	14362
BLR	163	3703
ratio	19	4

Gain de **19** en opérations se traduit par un gain de seulement **4** en temps: **comment faire mieux?**

Exemple de performances sur une matrice d'ordre **21 millions**

	opérations ($\times 10^{12}$)	temps (24 cœurs)
Référence	3119	14362
BLR→BLR+	163 → 48	3703 → 736
ratio	19 → 64	4 → 20

Gain de **19** en opérations se traduit par un gain de seulement **4** en temps: **comment faire mieux?**

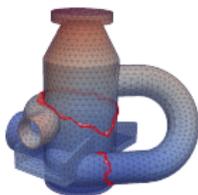
Améliorations algorithmiques (BLR+) nécessaires pour **mieux capturer le potentiel de réduction théorique en gains réels:**

- Gain en opérations encore plus important
- Meilleure efficacité: **15% → 25%** du pic de la machine atteint (mais encore loin du 80% de la méthode de référence!)

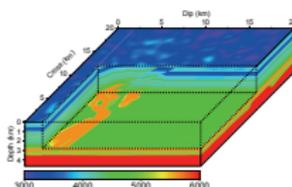


P. Amestoy, A. Buttari, J.-Y. L'Excellent, and T. Mary. *Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures*. ACM Trans. Math. Soft. (2018).

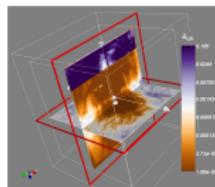
Résultats obtenus avec le logiciel **MUMPS** sur **900 cœurs**



Mécanique des structures
8M d'inconnues
Seuil de troncature 10^{-9}



Imagerie sismique
17M d'inconnues
Seuil de troncature 10^{-3}



Electromagnétique
21M d'inconnues
Seuil de troncature 10^{-7}

application	Temps de calcul (s)			Conso. mémoire (GB)		
	MUMPS	+BLR	ratio	MUMPS	+BLR	gain
mécastru	289.3	104.9	2.5	7.9	5.9	25%
sismique	617.0	123.4	4.9	13.3	10.4	22%
électromag.	1307.4	233.8	5.3	20.6	14.4	30%

Le BLR a permis de résoudre des **problèmes auparavant hors de portée!**

Exemple: problème d'imagerie sismique à haute fréquence
⇒ **130 millions d'inconnues** sur **2400 cœurs** (**150** → **4 PetaFlops**
et **25** → **11 TeraBytes** grâce au BLR), le tout **sans impacter la fiabilité**

Quels défis pour les solveurs exploitant des blocs de rang faible?

- **Problèmes de taille extrême:** pour passer à l'échelle sur des problèmes d'ordre 10^9 , besoin d'une meilleure complexité
⇒ comment atteindre une **complexité $\mathcal{O}(n)$ sans sacrifier la structure non hiérarchique**, idéale pour le calcul parallèle?
- **Architectures massivement parallèles:** pour passer à l'échelle sur 10K voire 100K cœurs, besoin de repenser les algorithmes
⇒ comment passer **de 25% à 80%** du pic machine atteint?
- **Fiabilité numérique:** l'erreur numérique mesurée est en général proche du seuil ε ⇒ **quelles garanties de fiabilité?** comment obtenir le meilleur **compromis entre précision et performance?**