RÉPUBLIQUE FRANÇAISE
*Liberté*
*Égalité*
*Fraternité*

# Factorisation LU sur des matrices à distribution non régulières

Projet de fin d'étude

Matteo Marcos

# Sommaire

1. Présentation
2. Contexte
3. Objectifs
4. Avancement
5. Conclusion

# 01
# Présentation

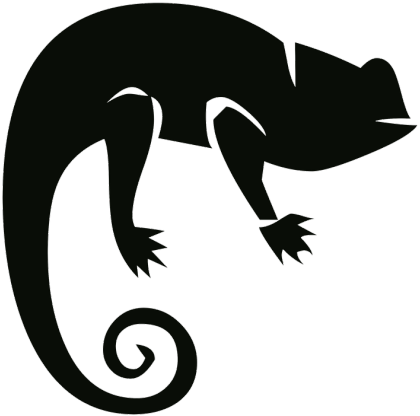
# Présentation

- 3 ans à l'ENSEIRB-MATMECA
- Spécialité Calcul Intensif et Science des Données (CISD)
- Projet de fin d'étude à l'INRIA dans l'équipe TOPAL

Encadrante : Alycia LISITO

# 02
# Contexte

# Présentation de Chameleon



**Bibliothèque d'algèbre linéaire dense en C**

Optimisé pour du calcul hétérogène et distribué

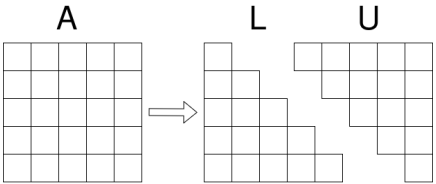
**Différentes précisions**

- Simple et double précisions
- Arithmétique réelle et complexe

**Différents runtimes**

**StarPU** / QUARK / PaRSEC / OpenMP

# Factorisation LU (GETRF)

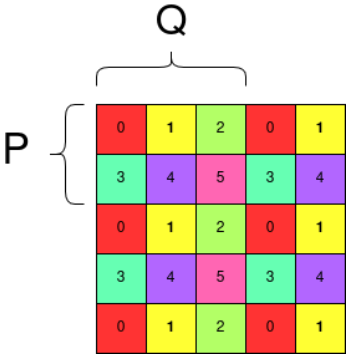



Factoriser une matrice en un produit d'une matrice triangulaire supérieure et inférieure

**2 méthodes implémentées**

- Sans pivotage
- **Avec pivotage partiel**

# Distribution 2D block cyclique (2DBC)




## 2 paramètres

- P : Nombre de processus par colonnes
- Q : Nombre de processus par lignes

✓ Bonne distribution sur des architectures homogènes, chaque processus a une quantité de travail similaire.

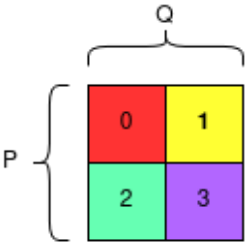
✗ Moins bonne sur des architectures hétérogènes, impossible d'ajuster les tâches aux capacités du noeud de calcul utilisé.

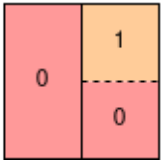
# Problème de réduction

Le GETRF nécessite des opérations de réductions
$\Longrightarrow$ Recherche du pivot

**en 2DBC**
Sachant P et Q, savoir quel processus est impliqué dans la réduction est automatique.

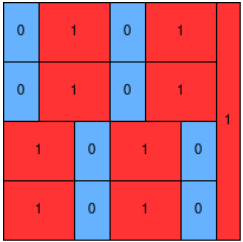
**Pour une distribution irrégulière**
Impossible de savoir automatiquement quel processus est impliqué.



# 03
# Objectifs

# Généralisation sur distributions irrégulières



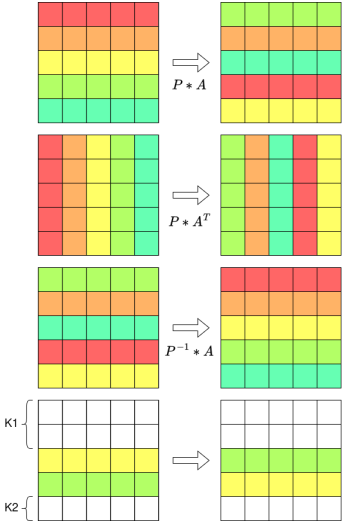
- Gérer l'accès aux identifiants des processus de manière efficace
- Adapter les algorithmes de réductions et les schémas de communications
- Comparer les performances avec la distribution 2DBC, et en utilisant les algorithmes de communications de OpenMPI et NewMadeleine

# 04
# Avancement

# Avancement

- ✓ Généralisation de la routine LASWP implémenté par Alycia dans le GETRF
- ✓ Implémentation des routines GESV et GETRS
- ✗ Optimisation des opérations de reduce du LASWP
- ✗ Implémentation du driver HPL pour le GETRF
- ✗ Gérer l'accès aux identifiants des processus de manière efficace
- ✗ Adapter les algorithmes de réductions et les schémas de communications
- ✗ Comparer les performances avec la distribution 2DBC, et en utilisant les algorithmes de communications de OpenMPI et NewMadeleine

# Implémentation du LASWP




Basée sur l'implémentation d'Alycia dans le GETRF

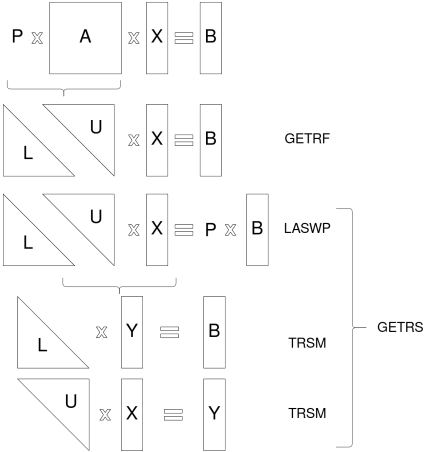
## Plusieurs fonctionnalités

- Permutation et permutation inverse pour le GESV.
- Permutation sur une partie de la matrice pour respecter le prototype de LAPACK.
- Permutation par colonne pour la thèse de Karmijn.

## Optimisation

✗ Changement du allreduce en reduce

# Implémentation du GESV et GETRS




## Implémentation de plusieurs routines autour du GETRF

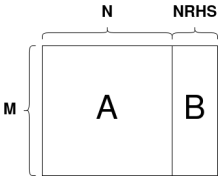
- GESV : General Solve
- GETRS : General Triangular Solve

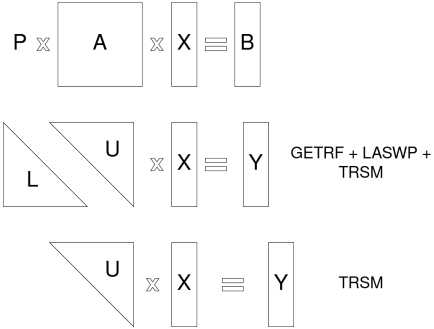
# Driver HPL pour le GESV

## Il est possible de regrouper certaines opérations

- Le GETRF fait un LASWP et un TRSM
- On peut en profiter pour les étendre sur B en même temps

## On modifie notre matrice d'entrée



# Conclusion

- ✗ Optimisation des opérations de reduce du LASWP
- ✗ Implémentation du driver HPL pour le GETRF
- ✗ Gérer l'accès aux identifiants des processus de manière efficace
- ✗ Adapter les algorithmes de réductions et les schémas de communications
- ✗ Comparer les performances avec la distribution 2DBC, et en utilisant les algorithmes de communications de OpenMPI et NewMadeleine

*Merci.*