



# Intégration du support de rang faible en mémoire distribuée dans PaStiX

# Sommaire

## 1. Contexte et représentation des données utilisées

Contexte de travail

Ordonnanceurs POSIX et support d'exécution

Représentation en rang plein

Représentation en rang faible

## 2. Réalisation

Implémentation pour les ordonnanceurs POSIX

Réduction des communications

Implémentation pour les supports d'exécution

Résultats

# Contexte de travail

## PaStiX

- ▶ Solveur d'algèbre linéaire creuse
- ▶ Support des machines hétérogènes distribuées

# Contexte de travail

## PaStiX

- ▶ Solveur d'algèbre linéaire creuse
- ▶ Support des machines hétérogènes distribuées

## GitLab

- ▶ Merge request
- ▶ Contrôle du code avec intégration continue

# Les ordonnanceurs

- ▶ Statique :
- ▶ Dynamique :
- ▶ Support d'exécution

# Les ordonnanceurs

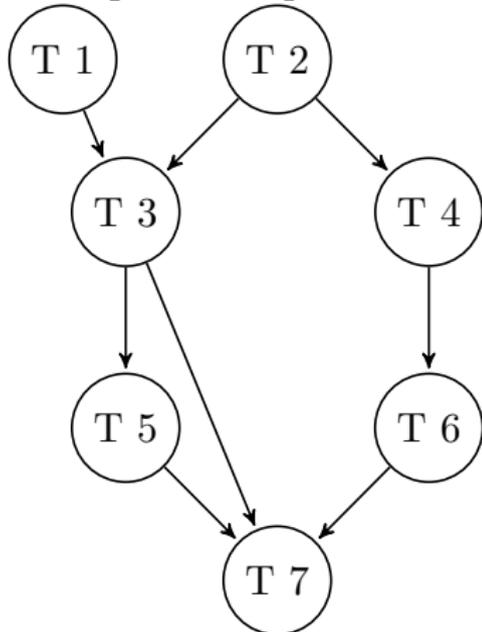
- ▶ Statique :
  - ▶ Répartition des tâches prédéterminées
  - ▶ Immuable durant l'exécution
- ▶ Dynamique :
  
- ▶ Support d'exécution

# Les ordonnanceurs

- ▶ Statique :
  - ▶ Répartition des tâches prédéterminées
  - ▶ Immuable durant l'exécution
- ▶ Dynamique :
  - ▶ Répartition initiale des tâches prédéterminées
  - ▶ Vol de travail permis
- ▶ Support d'exécution

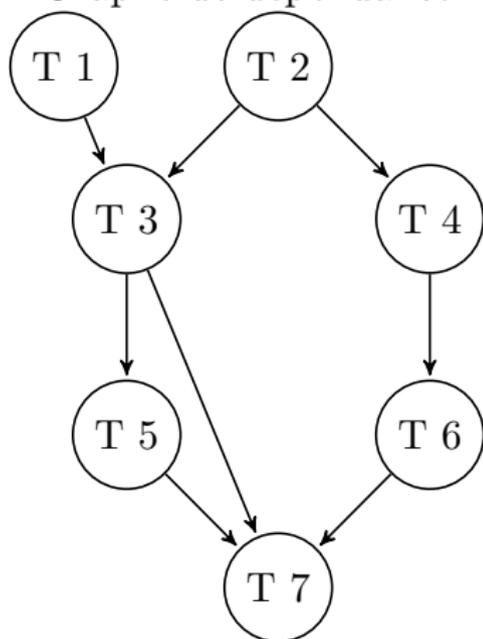
# Support d'exécution

Graphe de dépendance



# Support d'exécution

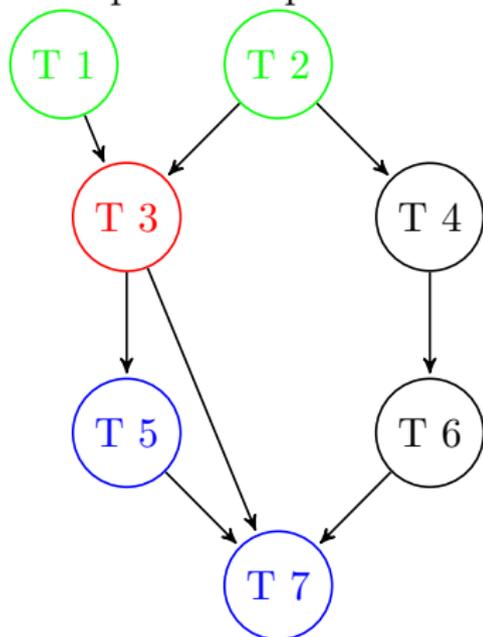
Graphe de dépendance



- ▶ StarPU : Vision globale des tâches et des dépendances

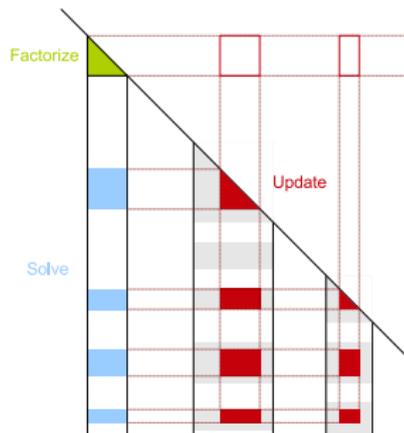
# Support d'exécution

Graphe de dépendance



- ▶ StarPU : Vision globale des tâches et des dépendances
- ▶ PaRSEC : Vision locale des tâches et des dépendances

# Représentation en rang plein



## Matrice

- ▶ Matrice creuse
- ▶ Colonnes  $\rightarrow$  blocs

## Représentation en rang faible

$$\begin{matrix} & \overbrace{\hspace{2cm}}^n \\ \underbrace{\hspace{1cm}}_m & \boxed{A} \end{matrix} \approx \begin{matrix} & \overbrace{\hspace{1cm}}^k \\ \underbrace{\hspace{1cm}}_m & \boxed{U} \end{matrix} \times \begin{matrix} & \overbrace{\hspace{2cm}}^n \\ \boxed{V^T} & \underbrace{\hspace{1cm}}_k \end{matrix}$$

Tel que  $\|A - U.V^T\| < \epsilon\|A\|$  avec  $\epsilon$  la précision souhaitée.

### Avantages

- ▶ Réduction de la consommation mémoire
- ▶ Réduction du temps de factorisation

# Sommaire

## 1. Contexte et représentation des données utilisées

Contexte de travail

Ordonnanceurs POSIX et support d'exécution

Représentation en rang plein

Représentation en rang faible

## 2. Réalisation

Implémentation pour les ordonnanceurs POSIX

Réduction des communications

Implémentation pour les supports d'exécution

Résultats

# Paquetage des données

## Ordonnanceurs POSIX

Rang plein

Rang faible

# Paquetage des données

Ordonnanceurs POSIX

Rang plein

1. Taille fixe

Rang faible

1. Taille variable

# Paquetage des données

## Ordonnanceurs POSIX

### Rang plein

1. Taille fixe
2. Données contigues

### Rang faible

1. Taille variable
2. Données non contigues  
→ Compactage des données

# Paquetage des données

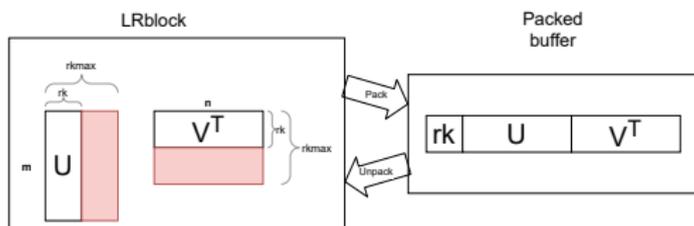
## Ordonnanceurs POSIX

### Rang plein

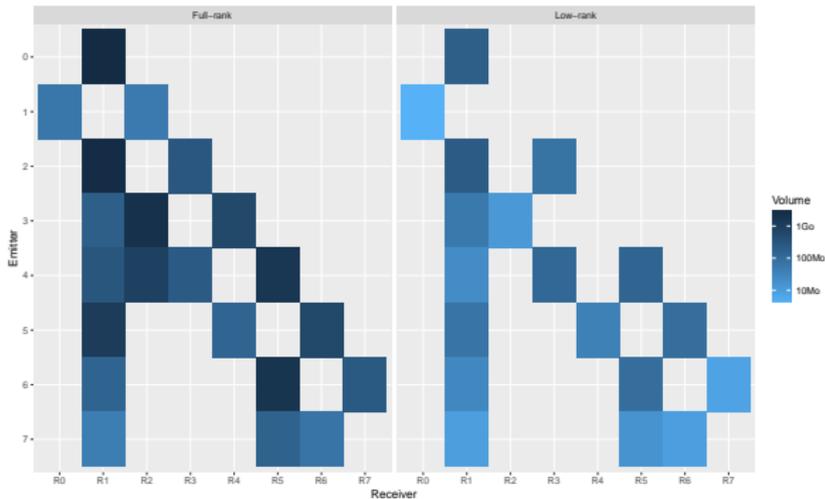
1. Taille fixe
2. Données contigues

### Rang faible

1. Taille variable
2. Données non contigues  
→ Compactage des données



# Réduction des communications



## Test

- ▶ Test sur 8 noeuds
- ▶ Comparaison entre rang plein/rang faible

# Implémentation pour les supports d'exécutions

- ▶ Besoins des supports d'exécutions

# Implémentation pour les supports d'exécutions

- ▶ Besoins des supports d'exécutions
  - ▶ Décrire les données

# Implémentation pour les supports d'exécutions

- ▶ Besoins des supports d'exécutions
  - ▶ Décrire les données
  - ▶ Déplacer les données

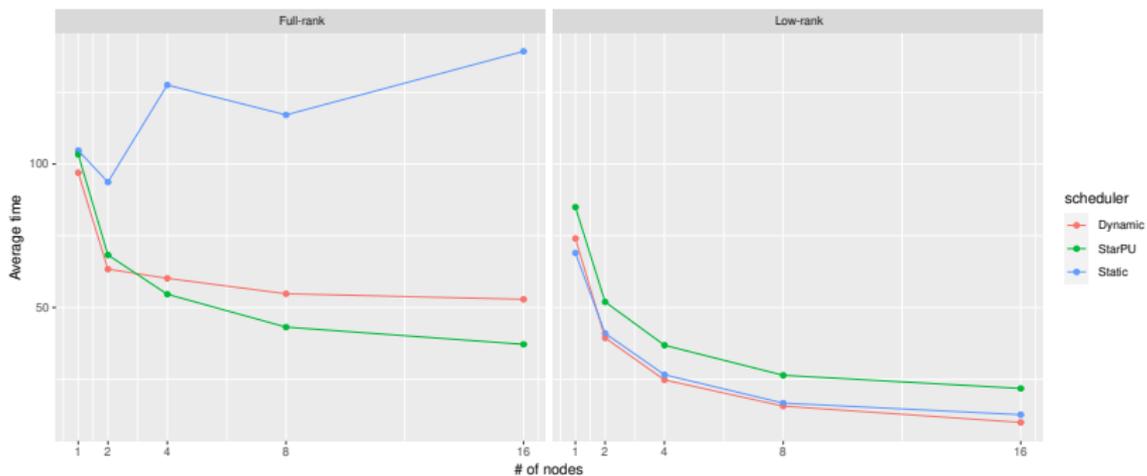
# Implémentation pour les supports d'exécutions

- ▶ Besoins des supports d'exécutions
  - ▶ Décrire les données
  - ▶ Déplacer les données
  - ▶ Allouer les données
  - ▶ Libérer les données

# Implémentation pour les supports d'exécutions

- ▶ Besoins des supports d'exécutions
  - ▶ Décrire les données
  - ▶ Déplacer les données
  - ▶ Allouer les données
  - ▶ Libérer les données
- ▶ Implémenter un filtre sur les données
  - ▶ Gestion plus fine

# Résultats



## En conclusion ?

- ▶ Le support de rang faible en mémoire distribuée est implémenté et donne des résultats intéressants

## En conclusion ?

- ▶ Le support de rang faible en mémoire distribuée est implémenté et donne des résultats intéressants
- ▶ Le support de rang faible n'est pas encore disponible pour PaRSEC

Merci pour votre attention.  
Avez-vous des questions ?