Transformations de programmes pour la grille

Stéphane Genaud

genaud@icps.u-strasbg.fr

LSIIT-ICPS, Université Louis Pasteur

Strasbourg – France

http://grid.u-strasbg.fr/

Introduction

- Le projet TAG (depuis septembre 2001)

Introduction

- Le projet TAG (depuis septembre 2001)
- Notre grille test

Introduction

- Le projet TAG (depuis septembre 2001)
- Notre grille test
- Deux applications de test

Introduction

- Le projet TAG (depuis septembre 2001)
- Notre grille test
- Deux applications de test
- Modifications envisageables

Introduction

- Le projet TAG (depuis septembre 2001)
- Notre grille test
- Deux applications de test
- Modifications envisageables
- Conclusion

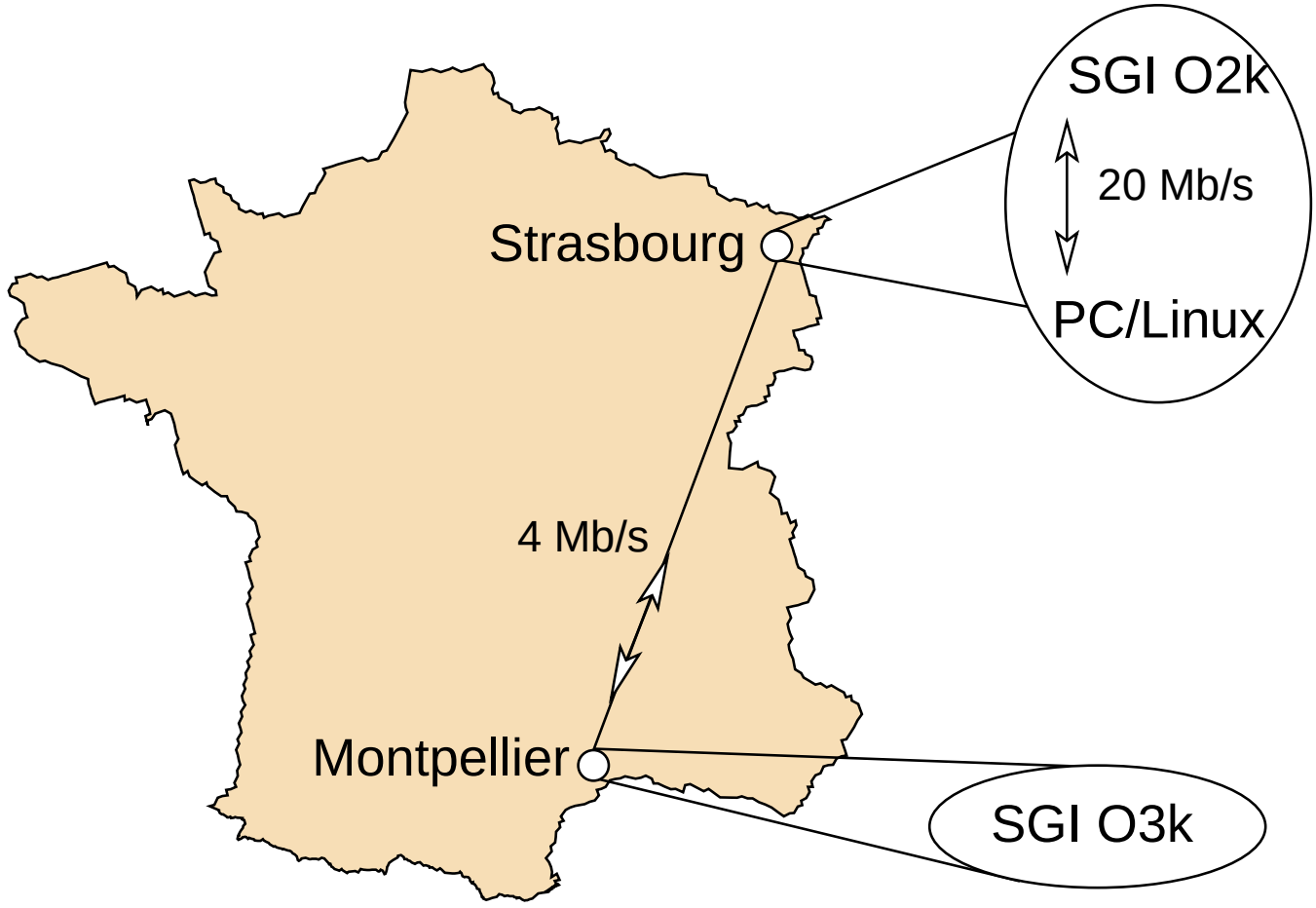
Le projet TAG

Objectifs du projet

- Acquérir une expertise sur l'exécution d'applications parallèles sur une grille
  - Codes scientifiques réels écrits en MPI
  - Grille de test basée sur Globus
- Credo: modification du code source nécessaire à l'équilibre de charge
- Apprendre de l'expérience, déduire des règles générales d'écriture de programme ou transformations

Une grille test

Notre grille expérimentale



Les nœuds de la grille

Nom	Localisation	Processeurs
seven	Illkirch	52×Mips
6 PC divers	"	Intel/AMD
pellinore	"	2×Intel
merlin	"	2×Intel
irmasrv2	Strasbourg (site Irma)	4×Sparc
irmasrv3	"	12×Sparc
2 PC divers	Strasbourg (site Eost)	1×Intel
PC	Strasbourg (site IECS)	1×Intel
biogrid01	Clermont-Ferrand	4×Intel
minerve	Montpellier	512×Mips

L’« intergiciel » installé

- Systèmes d’exploitation : Linux, IRIX, Solaris
- Globus (1.1.4, 2.0, 2.1)
- MPICH-G2 (1.2.2.3)

Points positifs

- MPICH-G2 permet l’exécution immédiate des applications MPI

Points positifs

- MPICH-G2 permet l’exécution immédiate des applications MPI
- MPICH-G2 s’interface avec la bibliothèque MPI du constructeur

Points positifs

- MPICH-G2 permet l’exécution immédiate des applications MPI
- MPICH-G2 s’interface avec la bibliothèque MPI du constructeur
- Globus gère les tâches d’identification (single sign-on)

Points positifs

- MPICH-G2 permet l'exécution immédiate des applications MPI
- MPICH-G2 s'interface avec la bibliothèque MPI du constructeur
- Globus gère les tâches d'identification (single sign-on)
- Globus facilite la localisation des binaires (GLOBUS_GASS_URL)

Points négatifs

- La compilation multi-systèmes est fastidieuse

Points négatifs

- La compilation multi-systèmes est fastidieuse
- La fiabilité d’une exécution est faible

Points négatifs

- La compilation multi-systèmes est fastidieuse
- La fiabilité d’une exécution est faible
- Rappel : Globus est un toolkit

Points négatifs

- La compilation multi-systèmes est fastidieuse
- La fiabilité d'une exécution est faible
- Rappel : Globus est un toolkit
  - N'intègre pas de système d'information sur l'état de la grille

Points négatifs

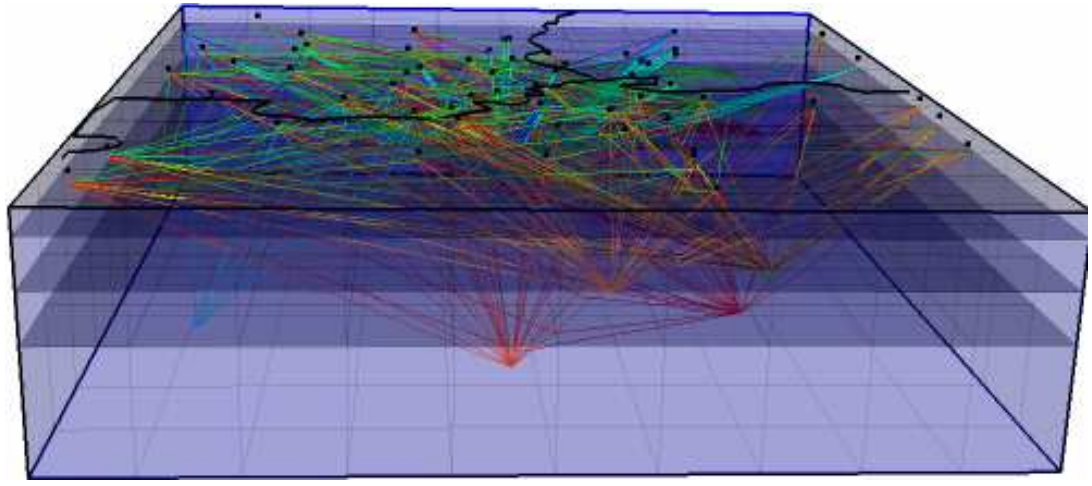
- La compilation multi-systèmes est fastidieuse
- La fiabilité d'une exécution est faible
- Rappel : Globus est un toolkit
  - N'intègre pas de système d'information sur l'état de la grille
  - Ne fournit pas d'ordonnanceur

Les applications

Code de géophysique

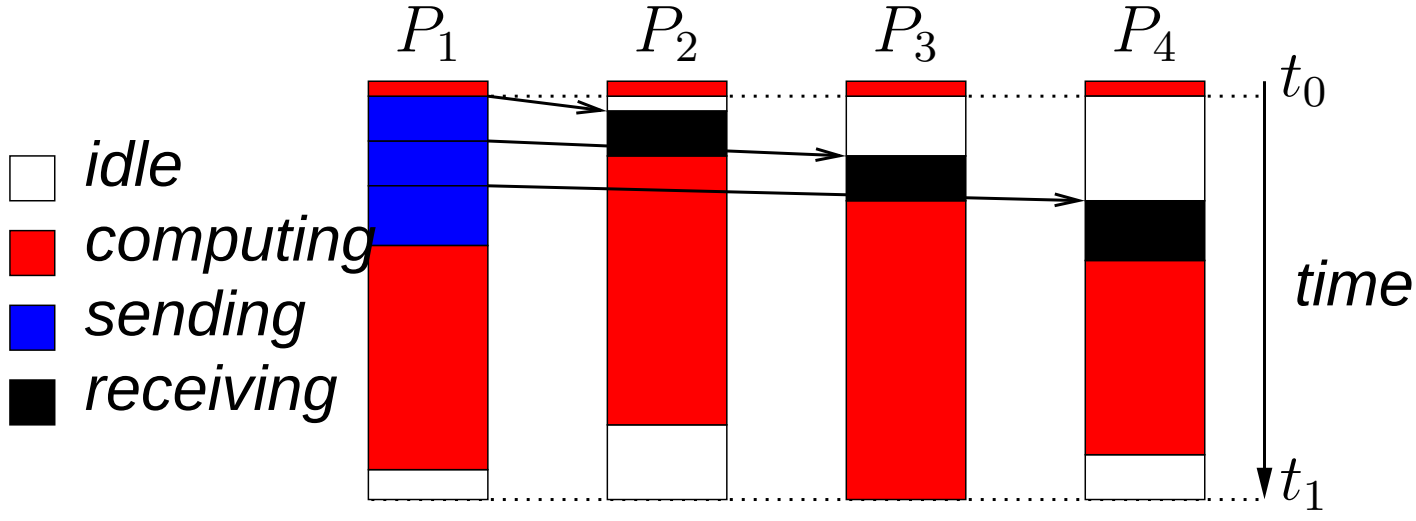
Tomographie sismique pour un modèle de vitesse

- Retracer les chemins des ondes sismiques enregistrées
- Mailler la zone étudiée (le globe entier)
- Calculer des variations de vitesse dans chacune des mailles



Code de géophysique

- Application *embarrassingly parallel* (phase 1)
- Distribution des données initiales : `MPI_Scatter`



- La distribution conduit à un « effet d'escalier »

Équilibre de charge

- On suppose :

Équilibre de charge

- On suppose :
  - Indépendance des données

Équilibre de charge

- On suppose :
  - Indépendance des données
  - Coût uniforme de traitement

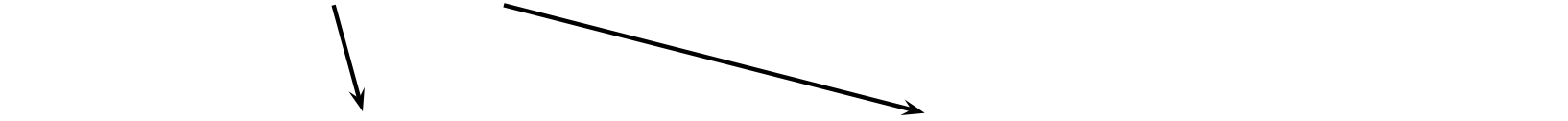
Équilibre de charge

- On suppose :
  - Indépendance des données
  - Coût uniforme de traitement
- Deux possibilités:

Transformer en maître/esclave dynamique

Équilibre de charge

- On suppose :
  - Indépendance des données
  - Coût uniforme de traitement
- Deux possibilités:



Transformer en maître/esclave dynamique

Remplacer `MPI_Scatter` par `MPI_Scatterv`

Équilibre de charge

Calcul de la nouvelle distribution:

- trouver la meilleure distribution (algo. général, lent)

Équilibre de charge

Calcul de la nouvelle distribution:

- trouver la meilleure distribution (algo. général, lent)
- programmation linéaire (contraintes linéaires, gourmand)

Équilibre de charge

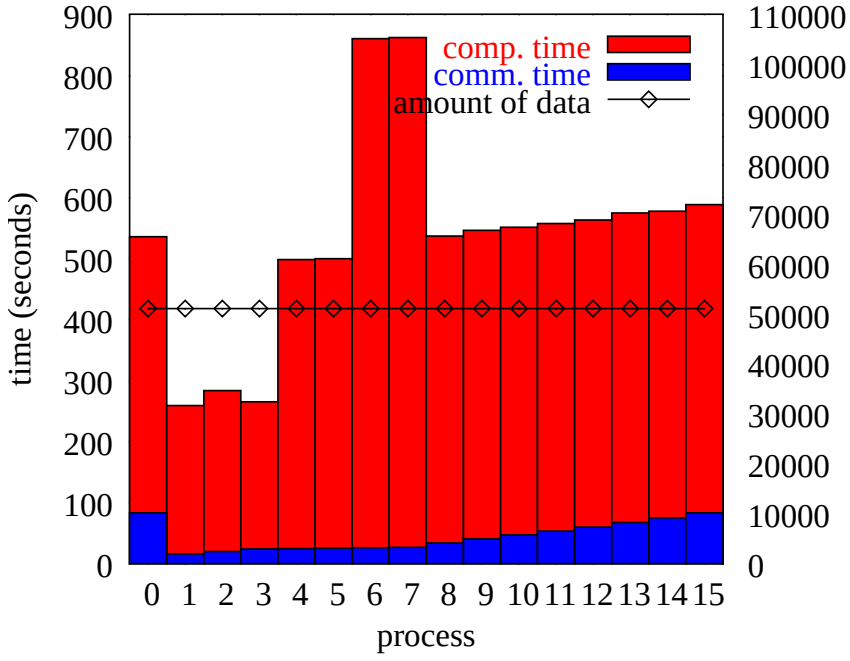
Calcul de la nouvelle distribution:

- trouver la meilleure distribution (algo. général, lent)
- programmation linéaire (contraintes linéaires, gourmand)
- approximation en rationnels d'un système d'équations (égalité des temps de fin) puis arrondi (rapide, suffisant)

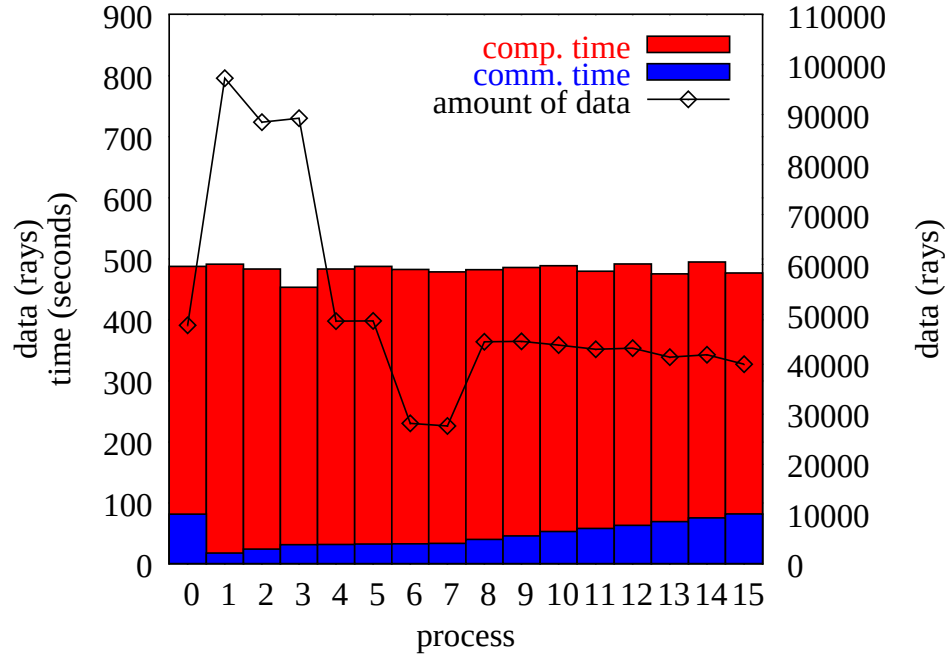
Expérience sur 16 processeurs

Nom	Localisation	Processeurs
dinadan	Strasbourg-Sud	1 × PIII/933
PC divers	“	4×XP1800+
pellinore	“	2×PIII/800
seven	Strasbourg-Sud	2×Mips R12K
minerve	Montpellier	8×Mips R14K

Résultats expérimentaux

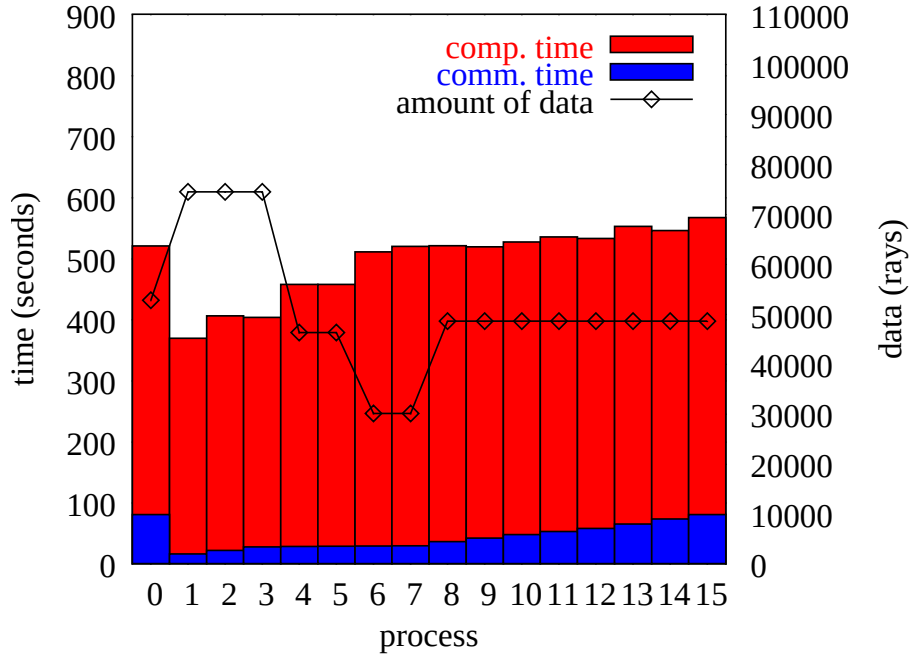


Sans équilibrage



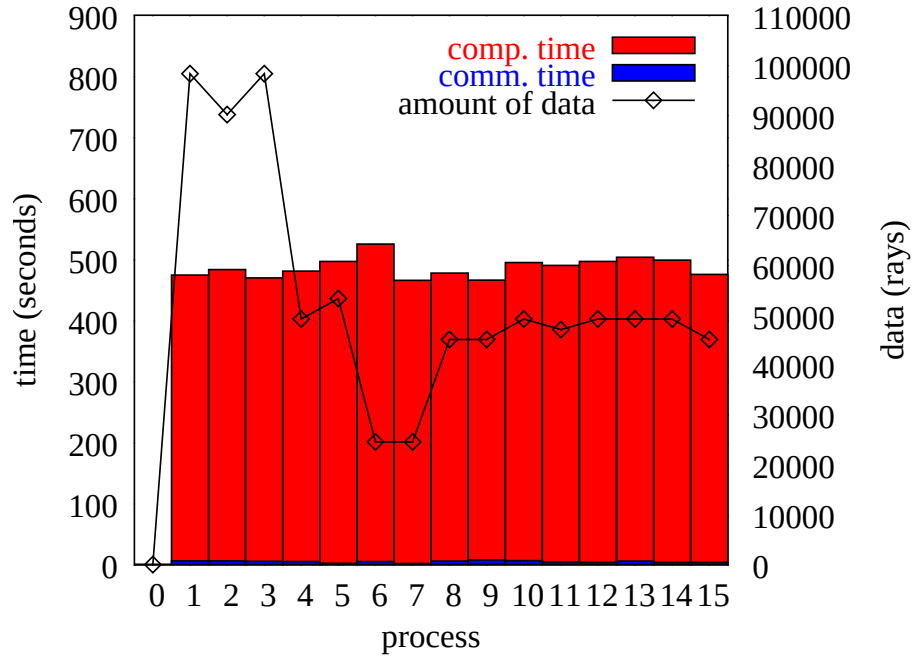
Équilibrage

Équilibrage sans réseau



Équilibrage sans réseau

Maître/Esclave

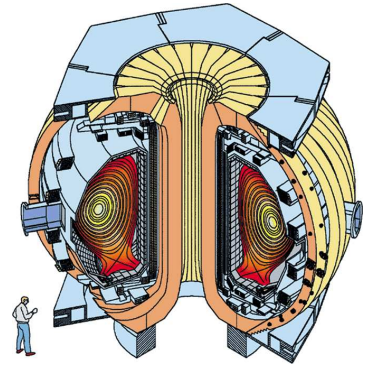


Maître/Esclave : nécessite de réécrire la partie communication

Physique des plasmas

Simulation de fusion nucléaire

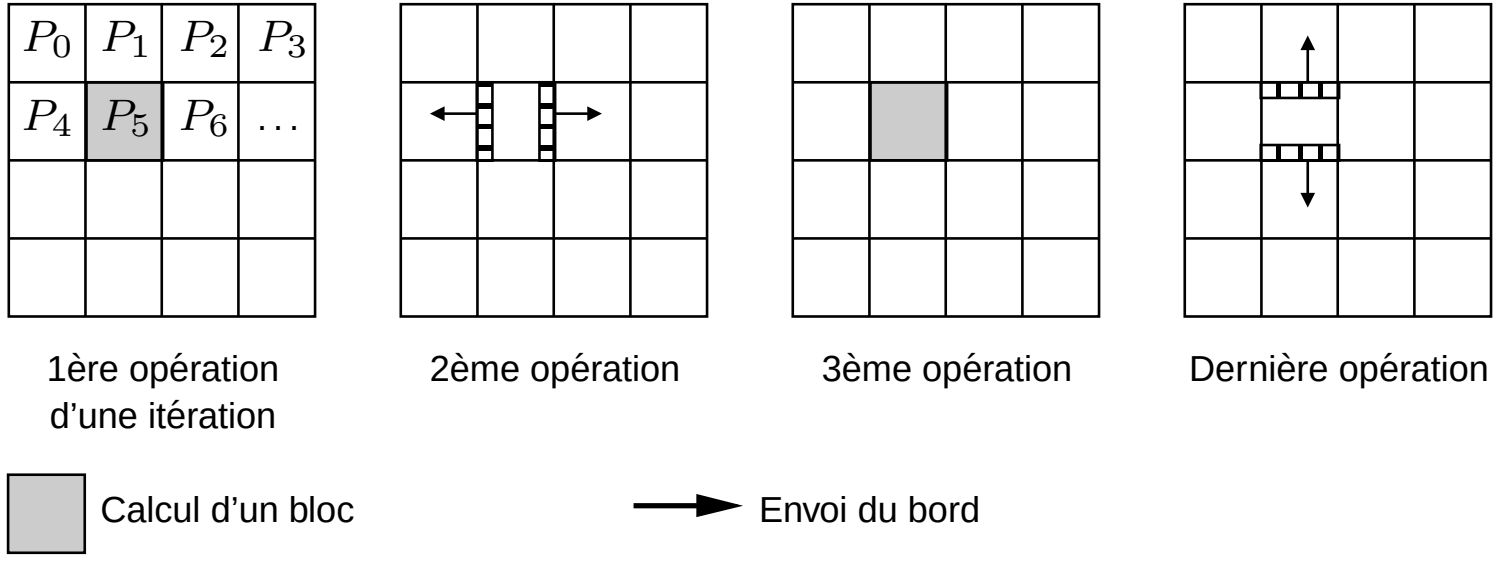
- Simulation non-particulaire dans une chambre de confinement (tokamak)
- Dans *l'espace des phases*, calcul de la probabilité de présence d'une particule



Vue éclatée d'un tokamak

Physique des plasmas

Suite d'itérations et de communications de proche en proche



Adaptation de l'application

But : permettre l'équilibre de charge dans l'application

- Distribution des données contrainte par les dépendances

Adaptation de l’application

But : permettre l’équilibre de charge dans l’application

- Distribution des données contrainte par les dépendances
- Nombre de processus libre

Adaptation de l'application

But : permettre l'équilibre de charge dans l'application

- Distribution des données contrainte par les dépendances
- Nombre de processus libre

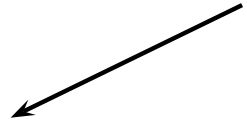
Solution : augmenter le nombre de processus/processeur

Adaptation de l'application

But : permettre l'équilibre de charge dans l'application

- Distribution des données contrainte par les dépendances
- Nombre de processus libre

Solution : augmenter le nombre de processus/processeur



- Immédiat sans réécriture

Adaptation de l'application

But : permettre l'équilibre de charge dans l'application

- Distribution des données contrainte par les dépendances
- Nombre de processus libre

Solution : augmenter le nombre de processus/processeur

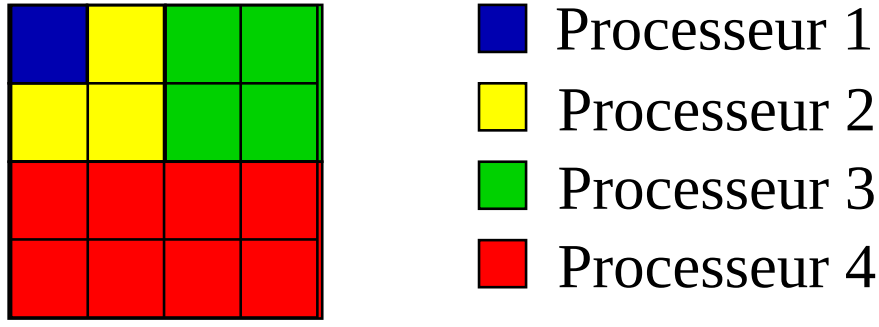
- Immédiat sans réécriture
- Réecriture : émulation de ce fonctionnement

Adaptation de l'application

Ajout d'une description du placement des blocs sur les processeurs. Le placement est fonction de la grille.

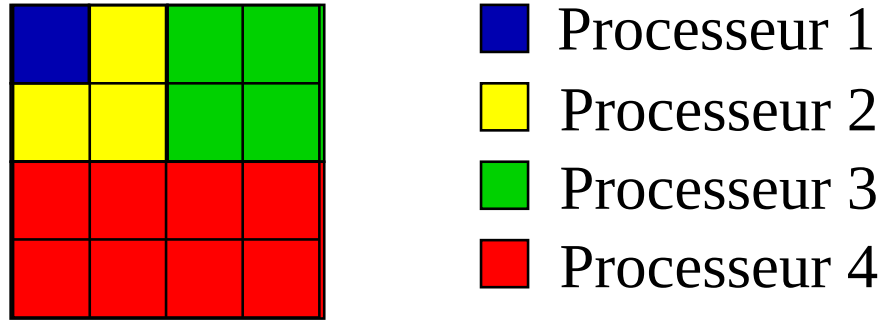
Adaptation de l'application

Ajout d'une description du placement des blocs sur les processeurs. Le placement est fonction de la grille.



Adaptation de l'application

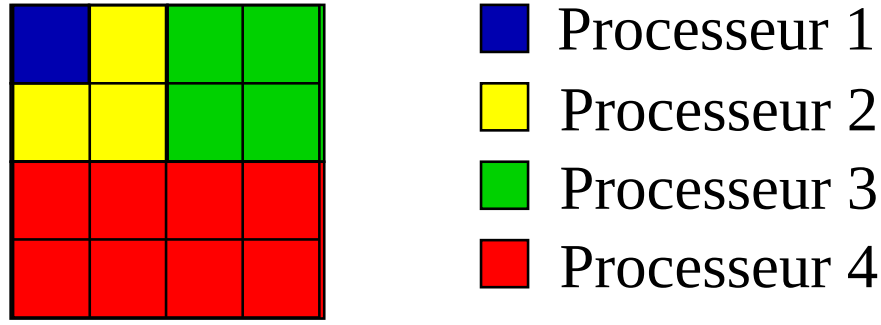
Ajout d'une description du placement des blocs sur les processeurs. Le placement est fonction de la grille.



Notion de *processus émulé*, définissant l'unité de travail élémentaire.

Adaptation de l'application

Ajout d'une description du placement des blocs sur les processeurs. Le placement est fonction de la grille.



Notion de *processus émulé*, définissant l'unité de travail élémentaire.

L'équilibrage de charge est fonction du nombre de processus émulés et du placement

Résultats

Comparaison du temps d'exécution :

- sur machine parallèle

Résultats

Comparaison du temps d'exécution :

- sur machine parallèle
- sur la grille sans puis avec équilibrage de charge

Résultats

Comparaison du temps d'exécution :

- sur machine parallèle
- sur la grille sans puis avec équilibrage de charge
- sur la grille en tenant compte de la topologie réseau

Résultats

Comparaison du temps d'exécution :

- sur machine parallèle
- sur la grille sans puis avec équilibrage de charge
- sur la grille en tenant compte de la topologie réseau
- sur la grille en minimisant les communications inter-processus

Configuration de la grille

Nom	Localisation	Processeurs
seven	Strasbourg-Sud	4×Mips R12K
PC divers	Strasbourg-Centre	2×PIV 1800
merlin	Strasbourg-Sud	2×XP 2000+
PC	“	1×XP 1800+
pellinore	“	2×PIII/800
athena	Montpellier	4×Mips R14K
PC	Strasbourg-Sud	1×XP 1800+

Résultats sur la grille

Configuration	CPU	Durée (s)
Grille sans eq.	16	1200
Grille avec eq.	"	1050
+ Topo	"	680
+ Répartition	"	500

Résultats sur machine parallèle

Configuration	CPU	Durée (s)
Machine parallèle	16	28
Machine parallèle	4	65

- Temps de communication très importants
- L'application est le prototype de la pire application pour une grille

Conclusions

MPI & grille, possible ?

- MPICH-G2 permet un portage immédiat
- Environnement Globus viable à court terme
- Exécution inefficace
- Nécessite au minimum d'équilibrer la charge
- Travail important ? Automatisable ?
- Aujourd'hui pas adapté à une petite granularité
- Examiner néanmoins le temps de restitution