Test de Primalité

Jean-Didier Pailleux - Robin Feron - Romain Robert - Damien Thenot - Maxence Joulin

UVSQ

13/05/2018

Introduction

- **Tests déterministes :** AKS, Pocklington, Euclide/Computation Bound, Eratosthène/Memory Bound.
- **Test probabiliste :** Miller-Rabin.
- **Rappel :** Miller-Rabin très efficace, méthodes naïves utiles pour les petits nombres et AKS peu performant à cause de NTL.
- **Objectif :** Optimisation des algorithmes et travail sur la parallélisation.

Introduction

Plan

Memory Bound

- Suppression des nombres pairs du tableau $\rightarrow$ Réduction de la taille du tableau.
- Réduction du nombre d'itérations $\rightarrow$ N à $\sqrt{N-3}$.

AKS

Implémentation d'une variante d'AKS $\rightarrow$ Complexité passant de $O(log(n))^{12}$ à $O(log(n))^3$.

Optimisation

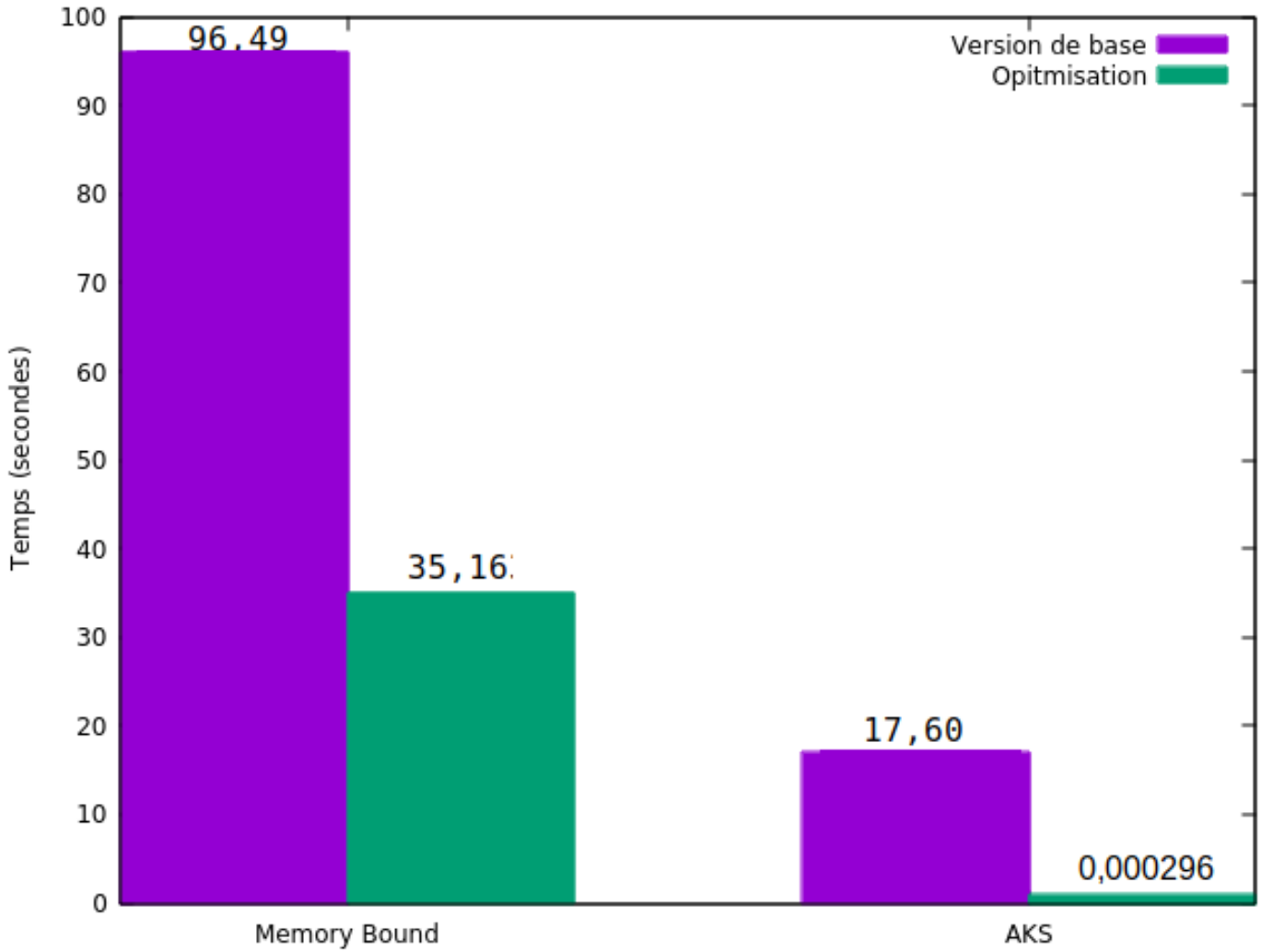


FIGURE – Comparaison de la version de base avec la version optimisé.

Plan

Outils utilisés

- OpenMP
- MPI (Message Passing Interface)

Outils utilisés

- OpenMP
- MPI (Message Passing Interface)

Parallélisation des algorithmes

- Implémentations non concluantes : AKS, Conjecture, Miller-Rabin, Pocklington
- Implémentation concluante : Memory Bound

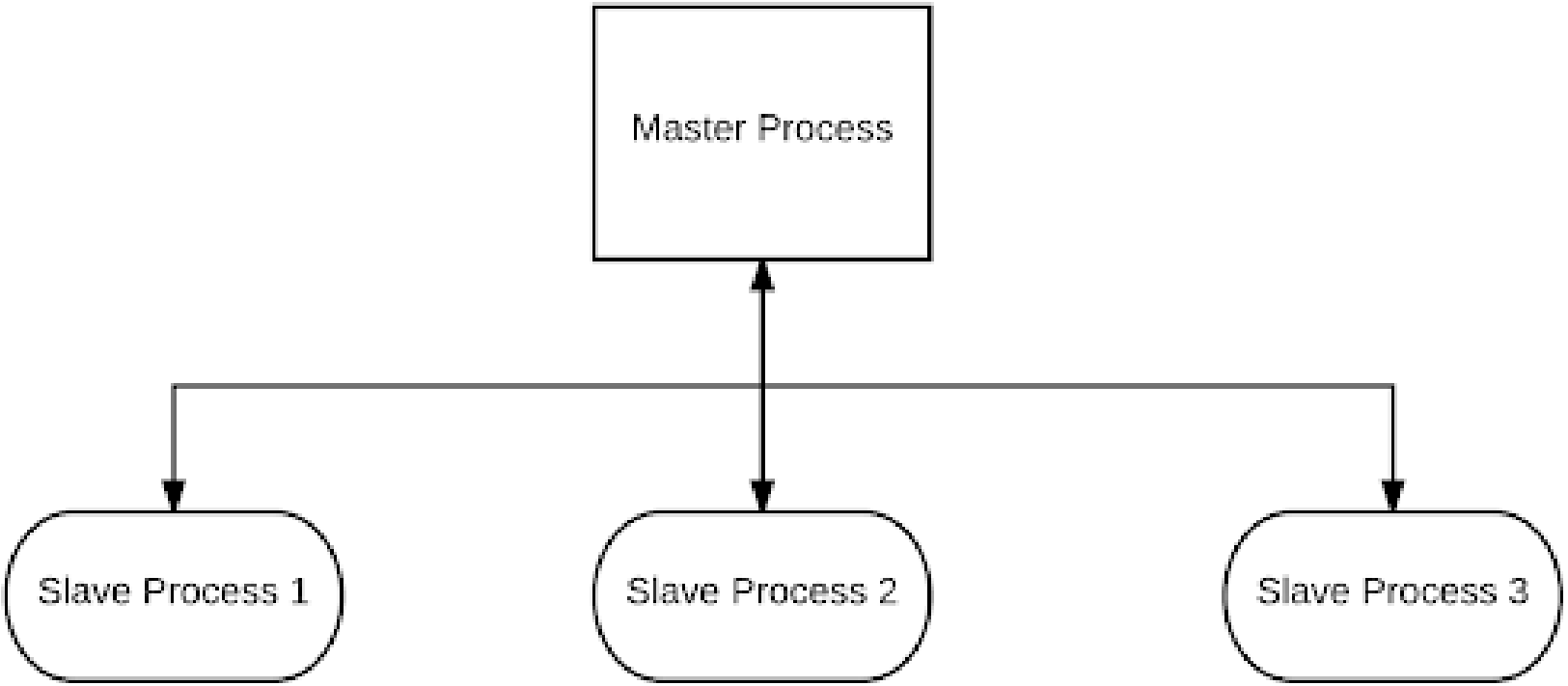


FIGURE – Schéma du Modèle du Master-Slave

Plan

Résultats

- Lancement des tests sur le cluster Poincare de la MDLS.
- Tests réalisés sur 10 itérations.
- Lors de la soumission du job Pocklington et AKS n'ont pas pu finir leur exécution avant d'être tué par le supercalculateur.

Résultats

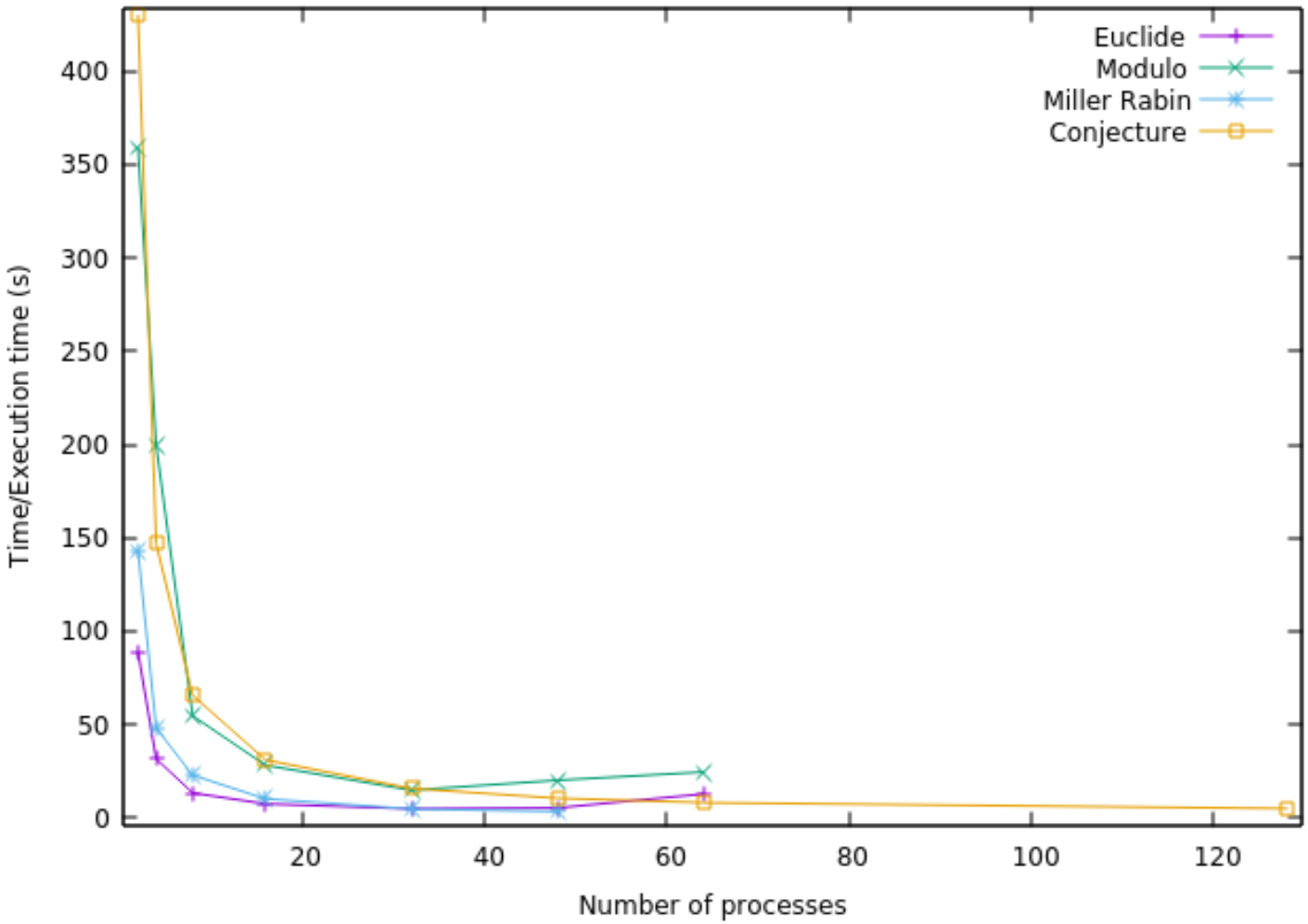


FIGURE – Évolution du temps de calcul pour Conjecture, Miller-Rabin, Euclide et Modulo ; Plage [1, 1 000 000]

Résultats

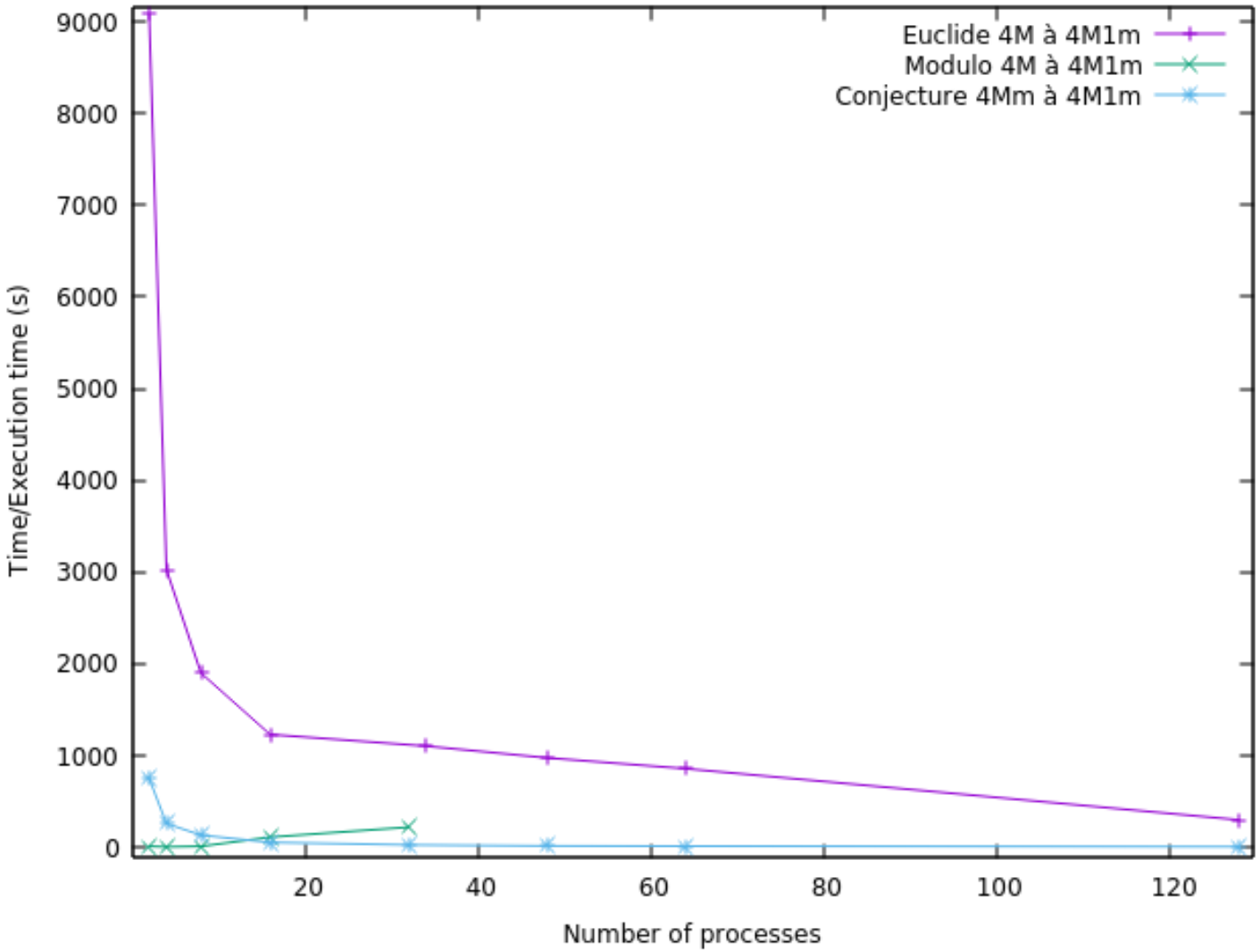


FIGURE – Évolution du temps de calcul pour Conjecture, Euclide et Modulo ; Plage [4M, 4M1m]

Résultats

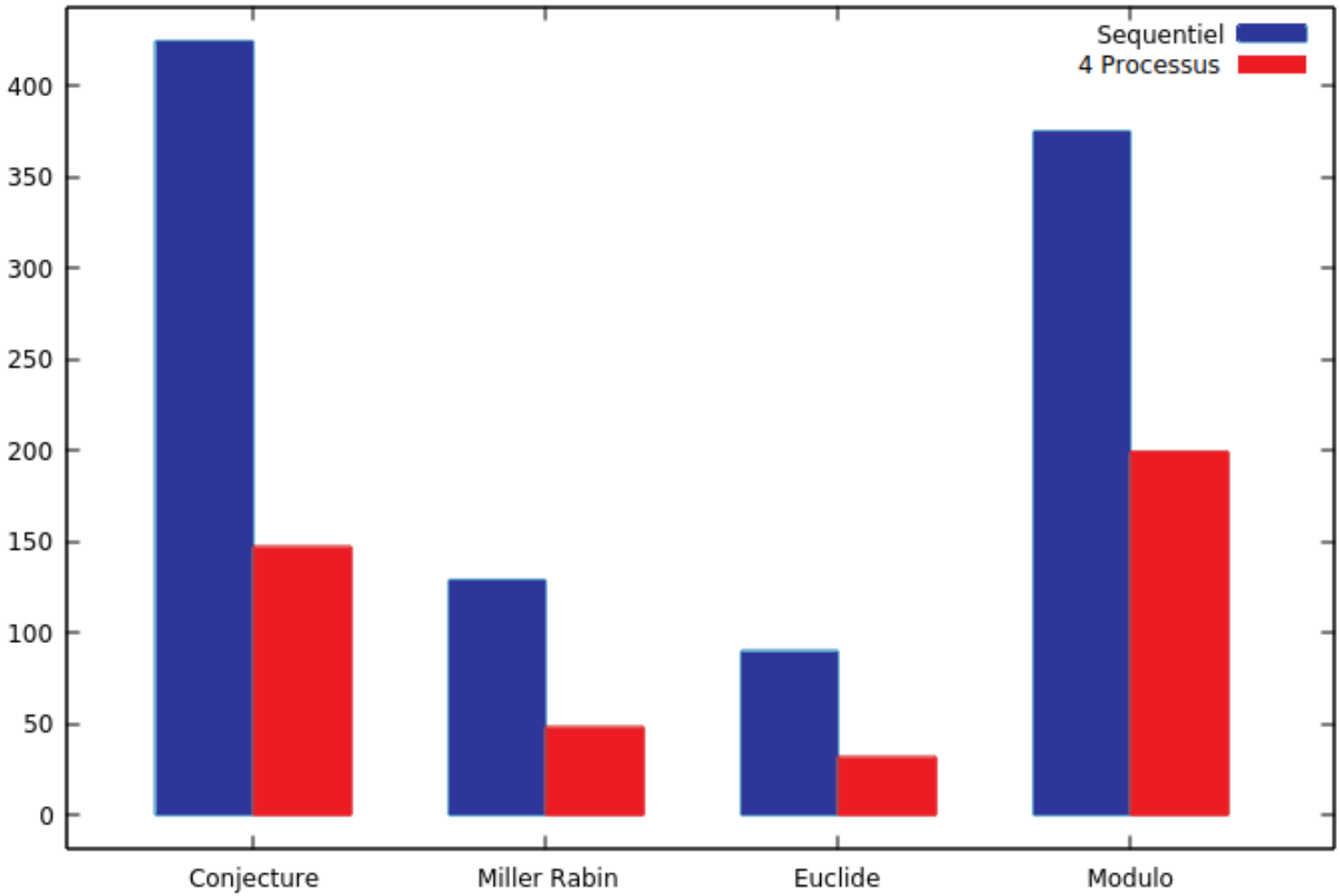


FIGURE – Comparaison Conjecture, Miller-Rabin, Euclide et Modulo avec le séquentiel

Résultats

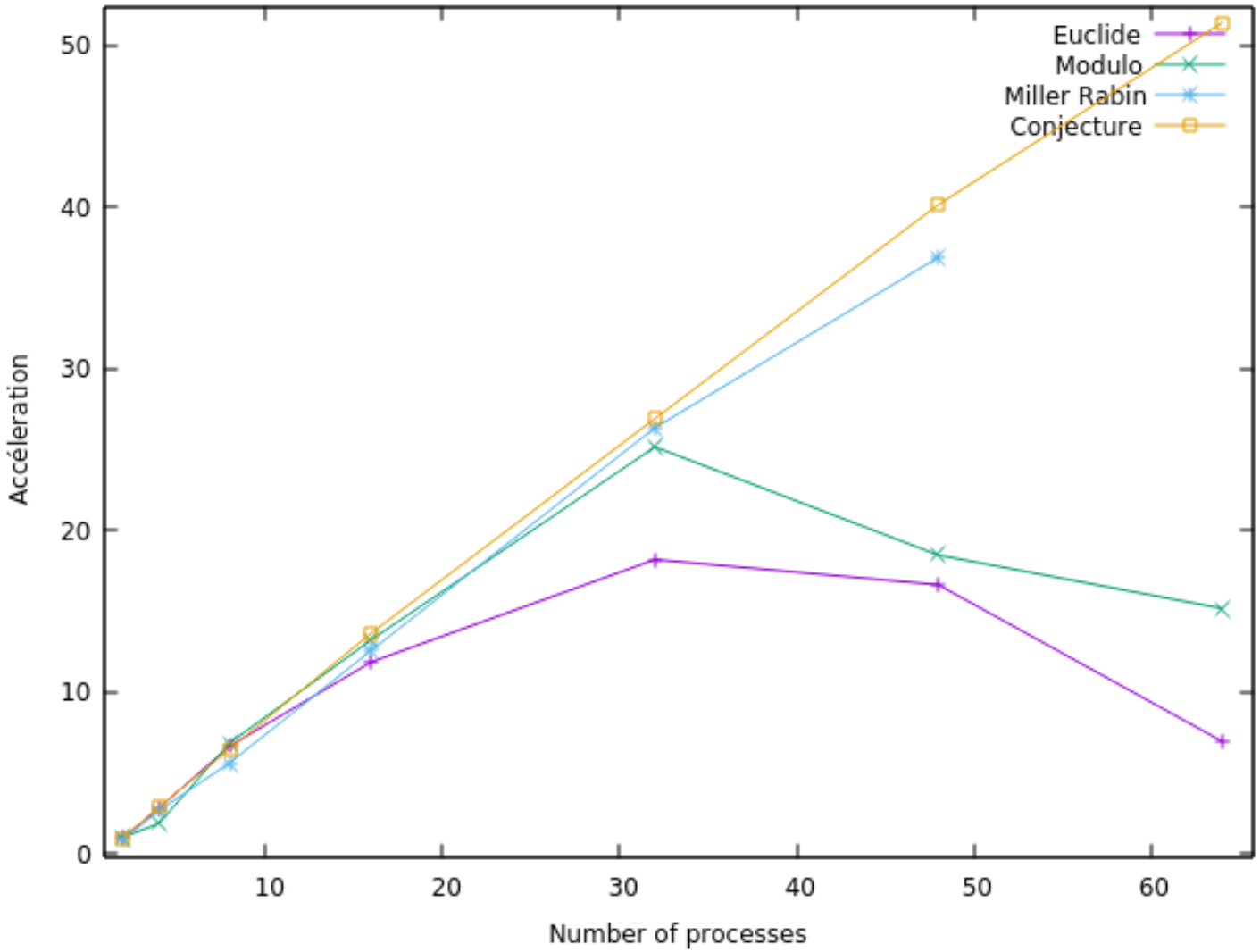


FIGURE – Évolution de l'accélération en fonction du nombre de processus ; Plage [1, 1 000 000].

Résultats

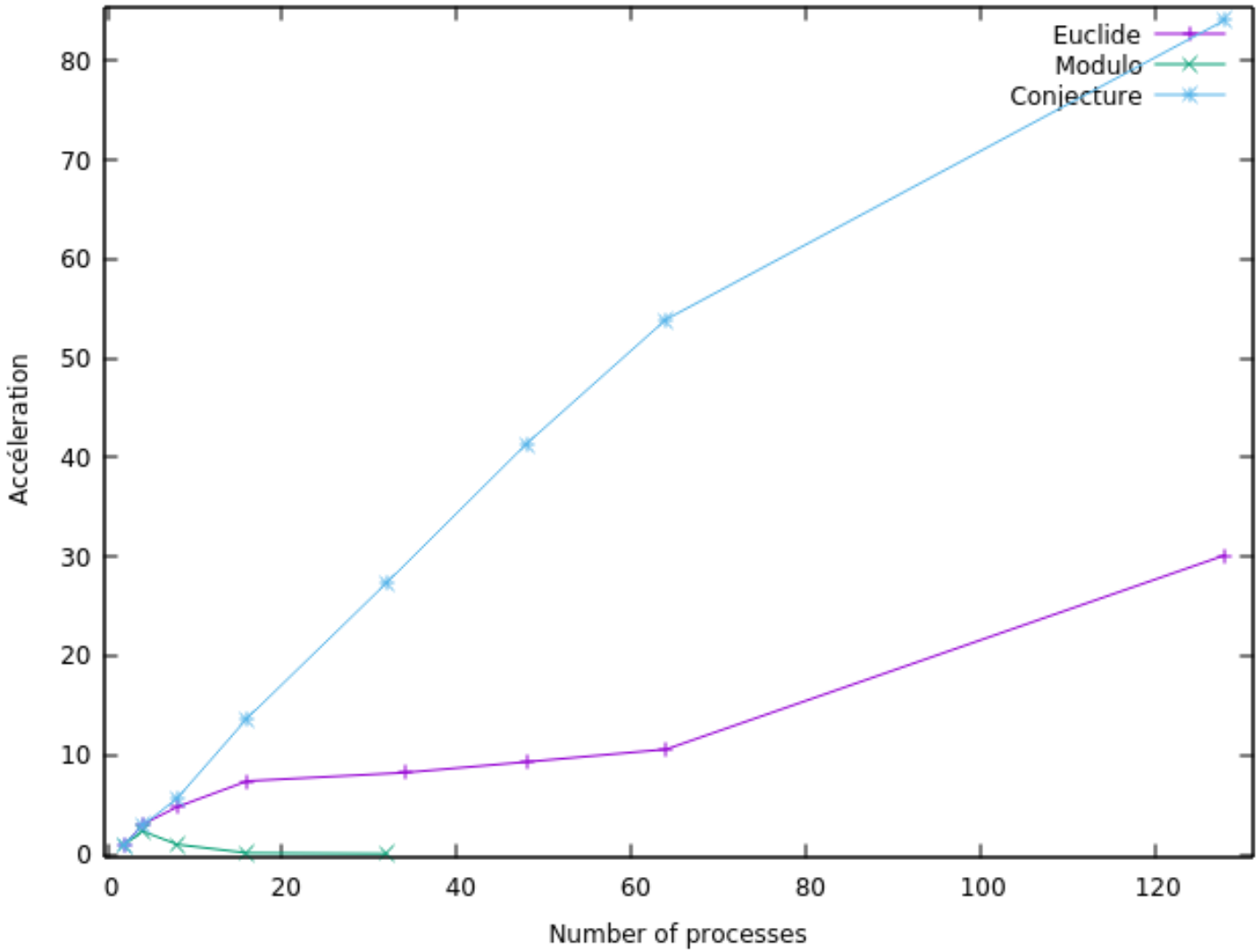
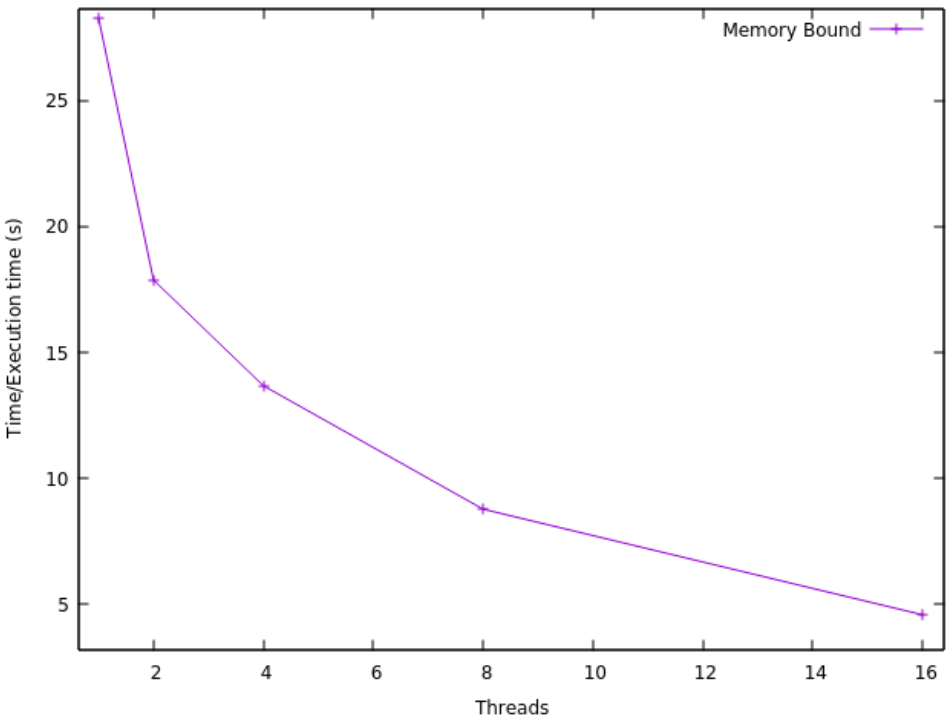
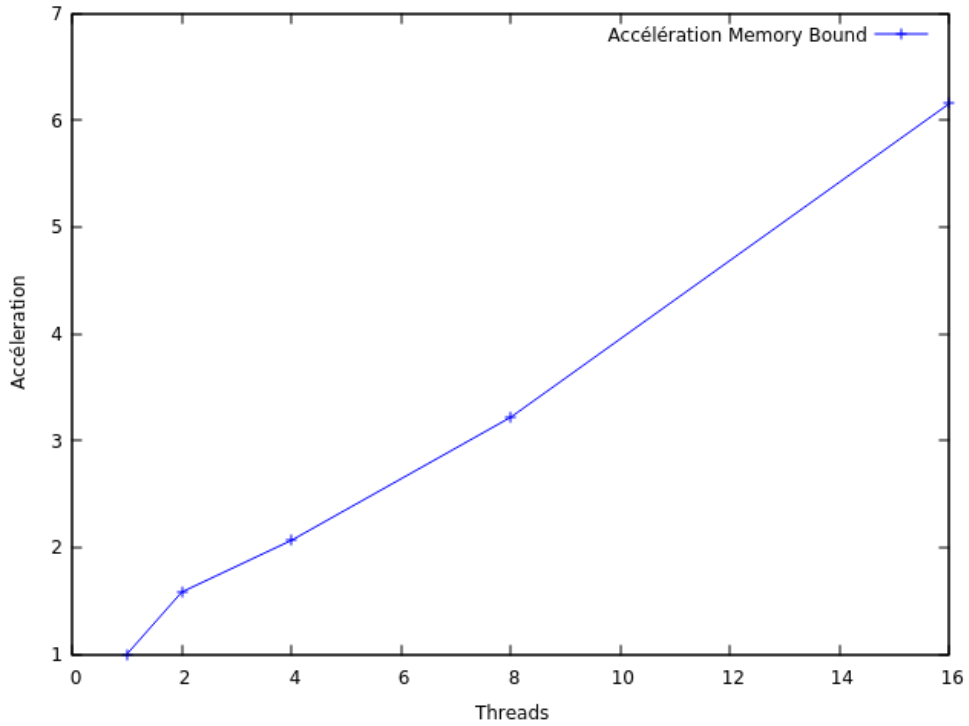


FIGURE – Évolution de l'accélération en fonction du nombre de processus ; Plage [4M, 4M1m].

Résultats





Figures - Évolution du temps de calcul et de l'accélération pour Memory Bound en fonction du nombre de threads.

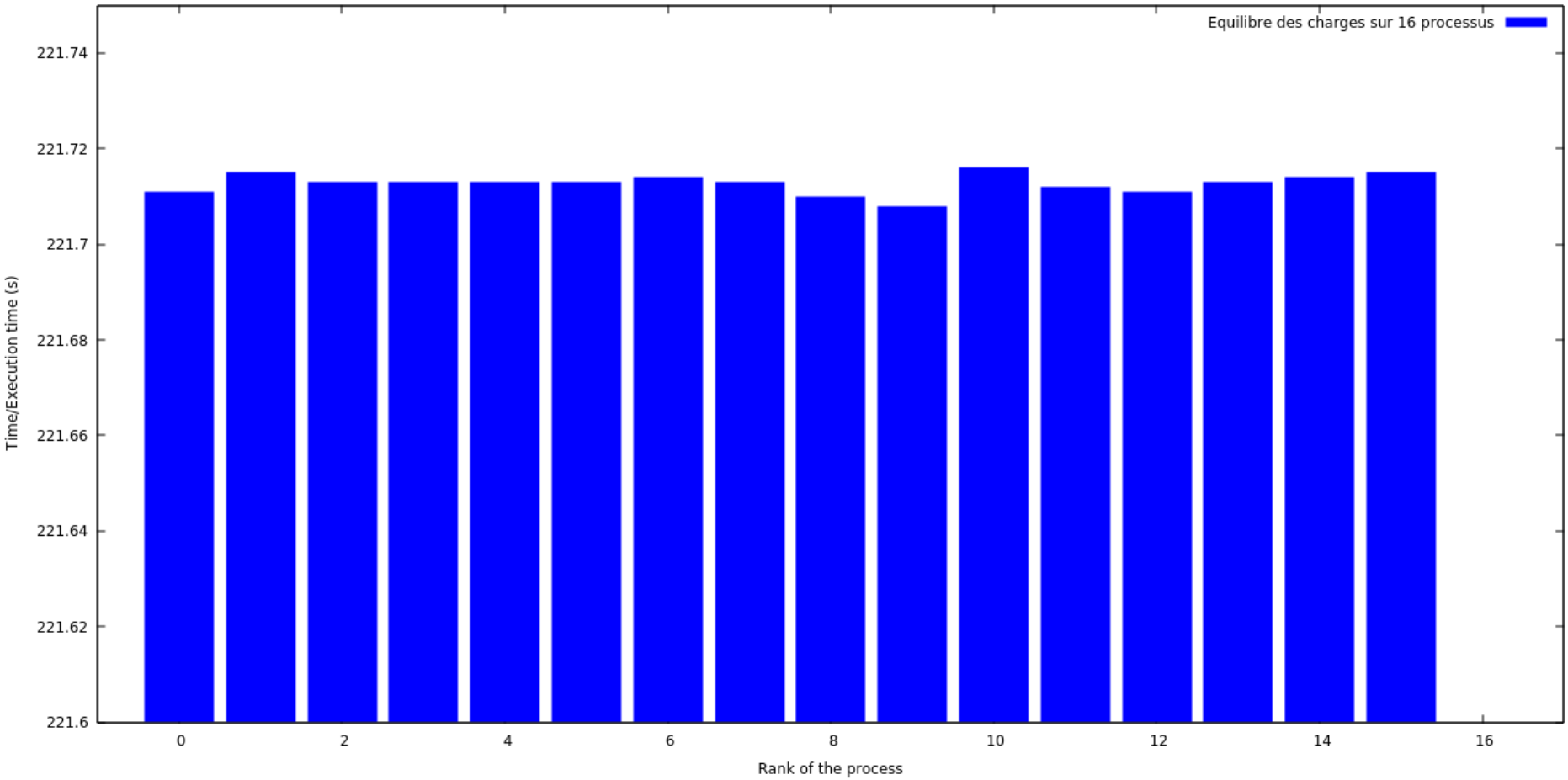


FIGURE – Exemple d'équilibre des charges obtenu avec 16 processus MPI.

Résultats

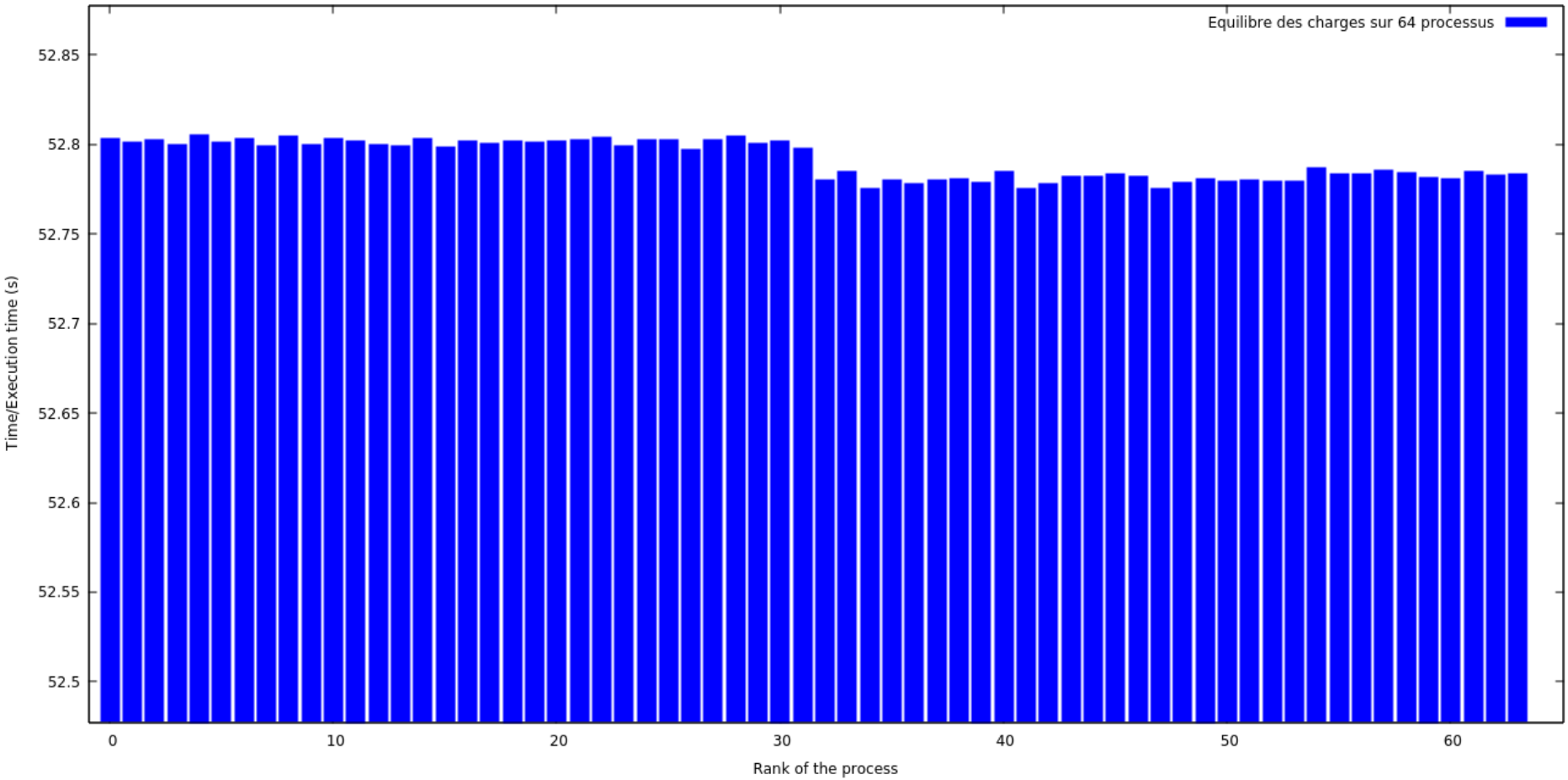


FIGURE – Exemple d'équilibre des charges obtenu avec 64 processus MPI.

Plan

Bilan Technique

- Une bonne scalabilité forte.
- Un palier max d'accélération.
- Le processus Maître trop chargé en travail.

De meilleurs schémas de communications

- Augmentation du nombre données à traiter par un esclave à chaque communication du Maître.
- Augmentation du nombre de Maître.

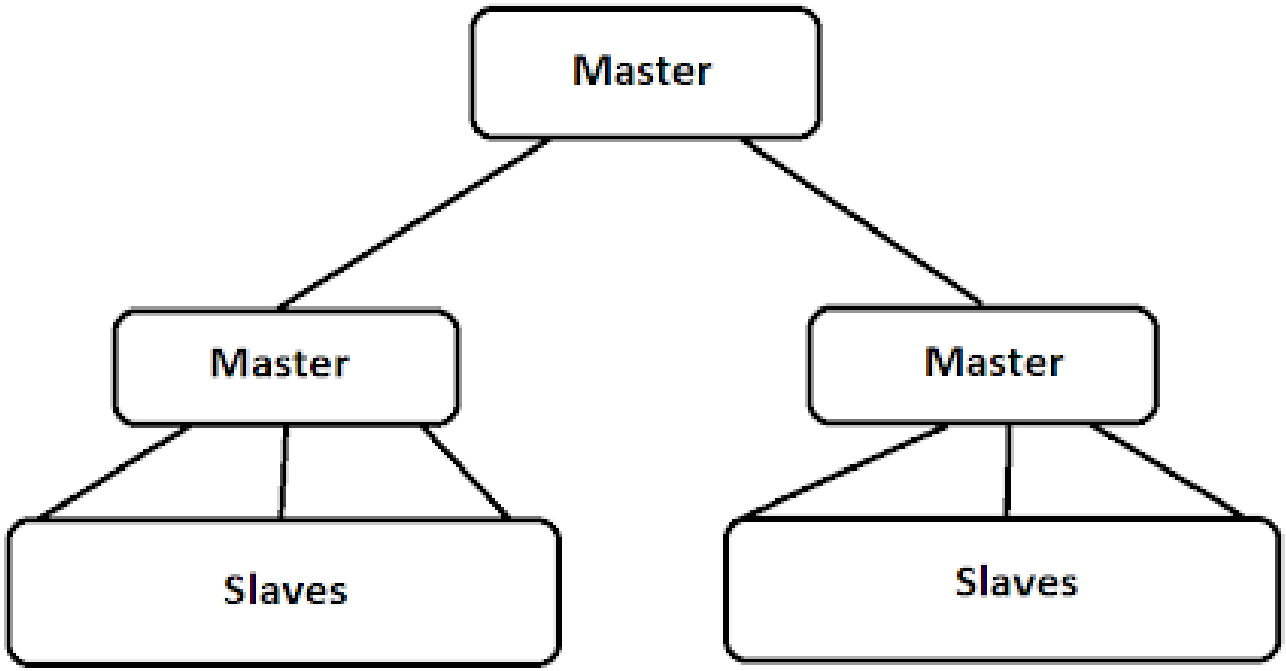


FIGURE – Schéma avec augmentation du nombre de Maître

Bilan Technique

Un équilibrage des charges efficace

- Des temps de calcul très courts.
- Des temps de calcul répartis de manière uniforme.
- Des différences finales négligeables entre les processus.

Bilan Technique

- Les tests de primalité sont peu parallélisable.
- Seule l'augmentation du volume de donnée permet de tirer avantage du parallélisme.

Plan

Problèmes Rencontrés

- Utilisation de bibliothèques sur le supercalculateur (NTL, GMP).
- Manque de temps.
- Dysfonctionnement du supercalculateur.

Plan

Conclusion

- Résultat final satisfaisant et équilibrage des charges efficace.
- Possibilité d'étudier d'autres techniques de parallélisation.
- Utilisation d'algorithmes nativement parallèle.