
Programmation Multicœur et GPU

Simulation de particules sur architectures parallèles

Auteurs :

Salmane BAH
Guillaume THIBAUT

Référents :

Raymond NAMYST
Pierre-André WACRENIER

Résumé

Ce rapport présente des mesures de performance portant sur une simulation d'atomes réalisée dans le cadre du projet de l'unité d'enseignement Programmation Multicœur et GPU. Dans cette partie, c'est plus particulièrement l'impact du tri des atomes qui y est mis en avant. Nous avons pour cela implémenté deux tris différents, un tri sur l'axe z, qui va traiter les atomes suivant leur position dans l'espace et un tri par boîte, plus optimisé, qui lui appliquera les forces seulement sur les atomes contenus dans les boîtes proches de l'atome traité.

1 Méthode d'expérimentation

Le script bash de la première expérimentation est réutilisé avec les mêmes paramètres d'expérimentation. Dans chacun de cas, les courbes d'accélération sans et avec tri sont présentées pour illustrer l'impact du tri.

2 Tri selon l'axe Z

Cette section présente l'impact du tri des atomes suivant l'axe z sur la performance de la simulation.

2.1 Type de machine

Cette expérience a été réalisée sur la machine Tchekov (salle 203) et sur le serveur Boursouf. Le nombre d'atomes utilisé est de 200 pour 100 iterations.

Les courbes

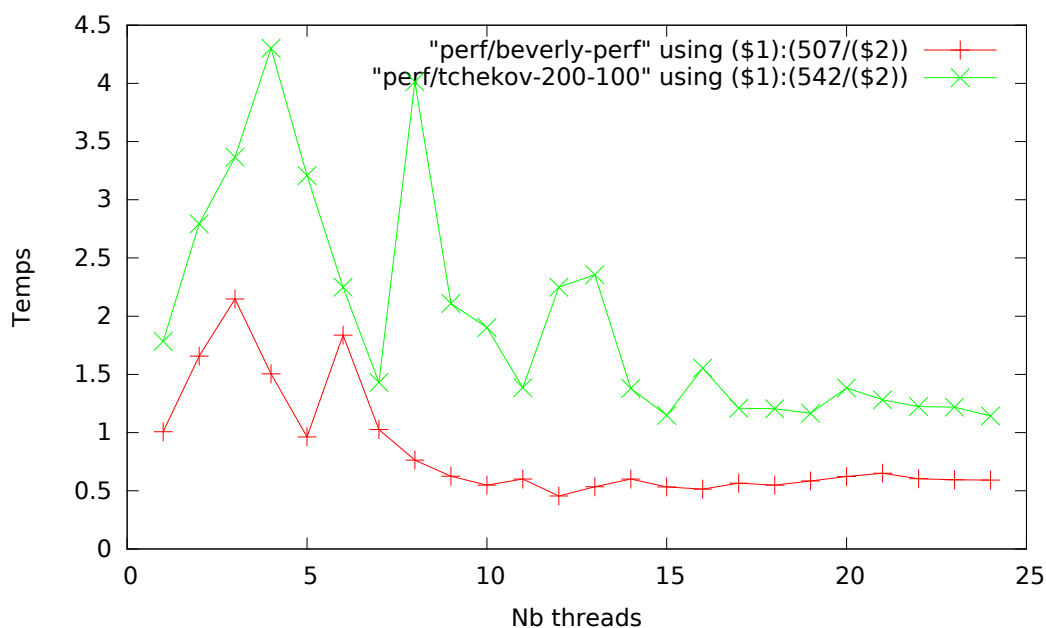


FIGURE 1 – Expérience salle 203, impact du tri selon l'axe z

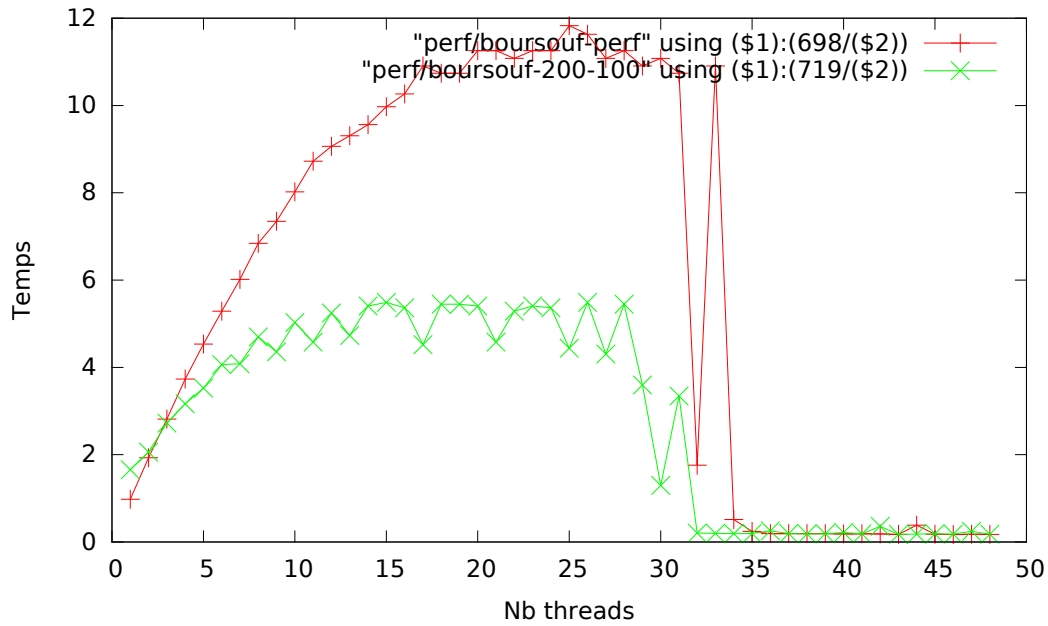


FIGURE 2 – Expérience sur Boursouf, impact du tri selon l'axe z

Commentaire

On voit une nette amélioration de l'accélération sur la machine Tchekov, le tri a réduit considérablement la charge des threads par unité d'itération. Cependant, sur le serveur Boursouf, on perd en performance et il est difficile d'expliquer pourquoi.

2.2 Placement des threads et de la mémoire

Cette expérience a été réalisée sur la machine Tchekov (salle 203) et sur le serveur Boursouf. Le nombre d'atomes utilisé est de 200 pour 100 iterations.

Les courbes

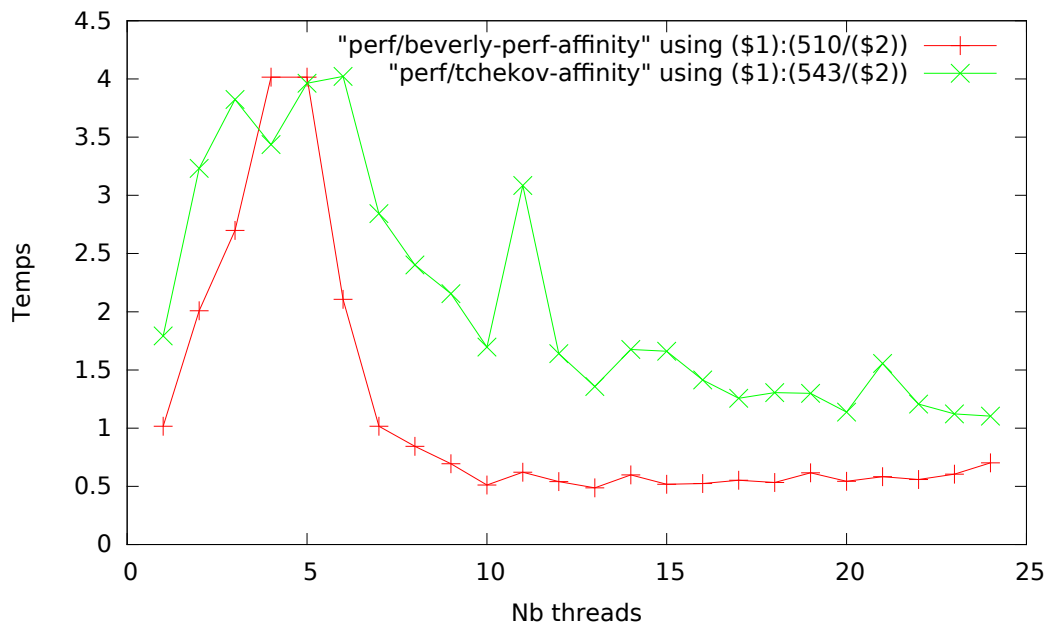


FIGURE 3 – Expérience salle 203, impact de l'affinité et du tri selon l'axe z

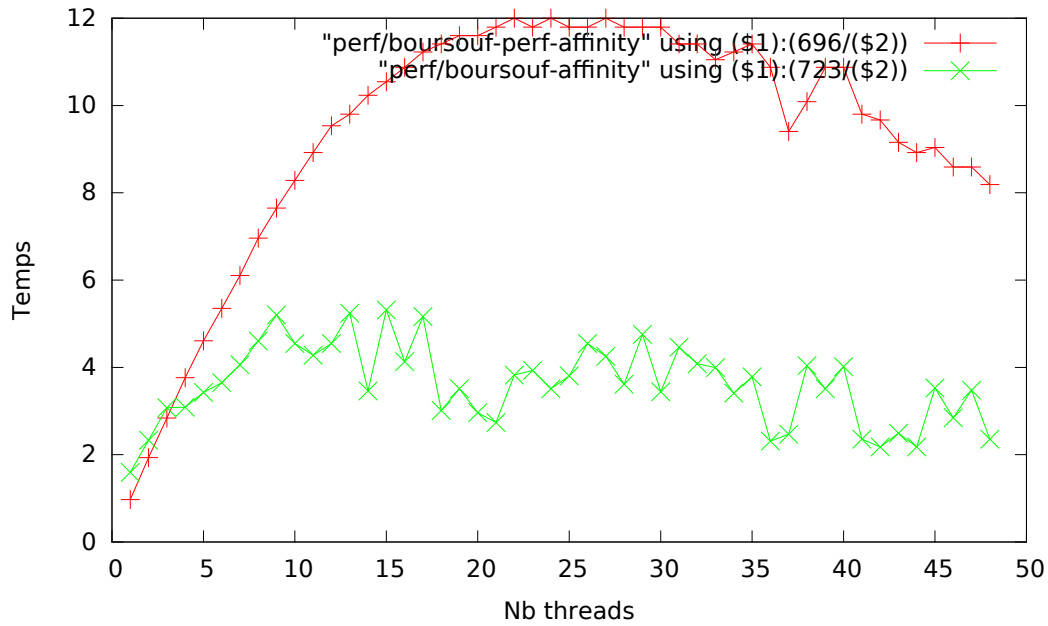


FIGURE 4 – Expérience sur Boursouf, impact de l’affinité et du tri selon l’axe z

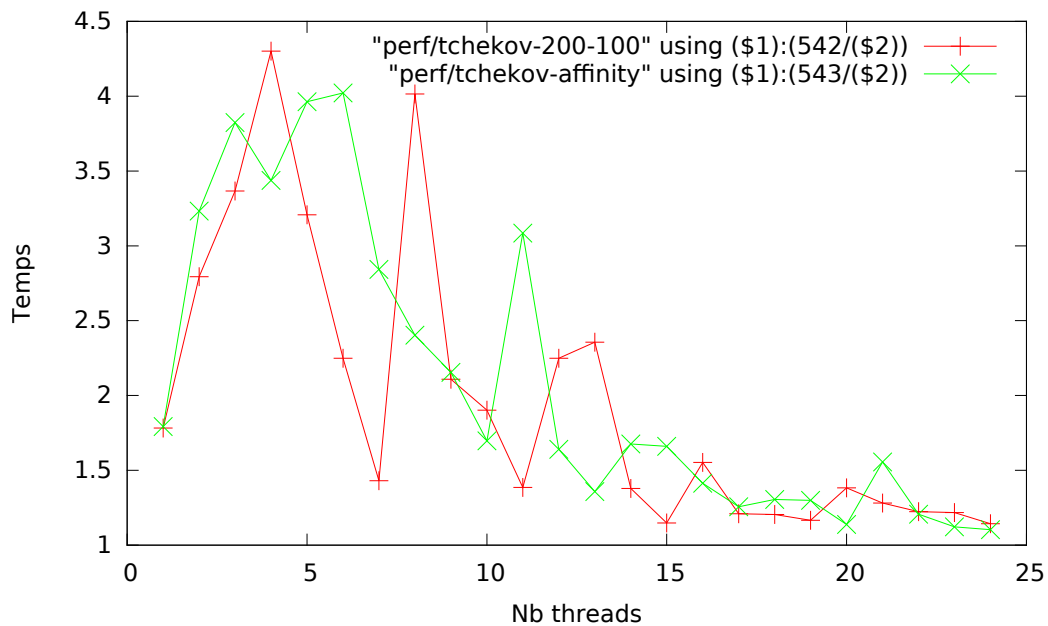


FIGURE 5 – Expérience salle 203, impact de l’affinité avec le tri sur l’axe z

Commentaire

Sur Tchekov (salle 203), comme pour l’expérimentation précédente, la version triée présente une meilleure accélération. L’accélération résultant de l’affinité des threads, elle, est négligeable.

Nous pouvons émettre l’hypothèse suivante : l’affinité permet à un thread préempté par l’OS de réutiliser des restes de son cache de son dernier passage. Dans la version triée, le cache d’un thread n’est pas assez volumineux (le thread ne regarde que les boîtes voisines), il y’a donc peu de chances pour qu’il y est des restes de ce cache la prochaine fois que le thread est élu. Dans ce cas, c’est comme si le thread avait changé de processeur, il doit recharger ses données depuis la mémoire (défaut de cache initial), d’où la similitude des courbes de performance sur la figure 5.

2.3 Politique de distribution d'indice

Les expérimentations suivantes ont été réalisées avec 1000 atomes et 100 itérations.

Stratégie de distribution statique

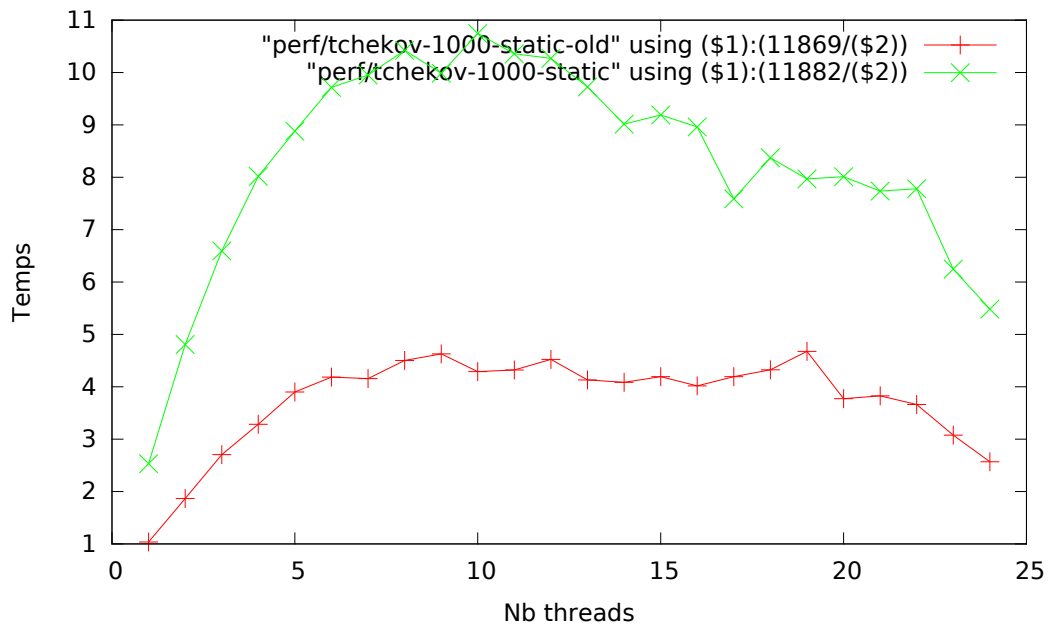


FIGURE 6 – Expérience sur Tchekov, impact de la politique statique et du tri

Stratégie de distribution cyclique

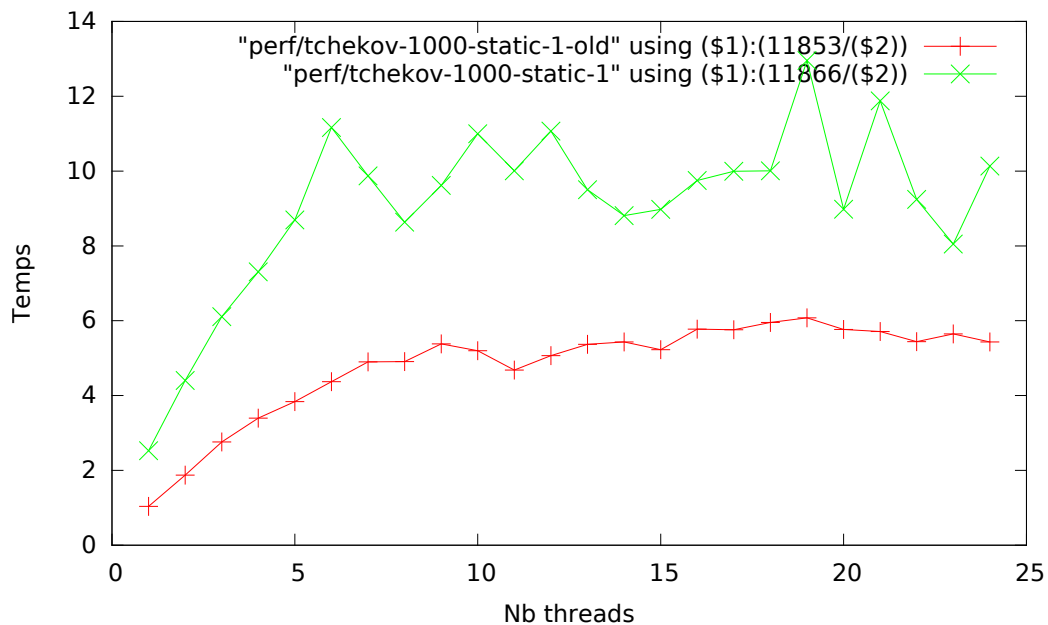


FIGURE 7 – Expérience sur Tchekov , impact de la politique cyclique et du tri

Stratégie de distribution dynamique

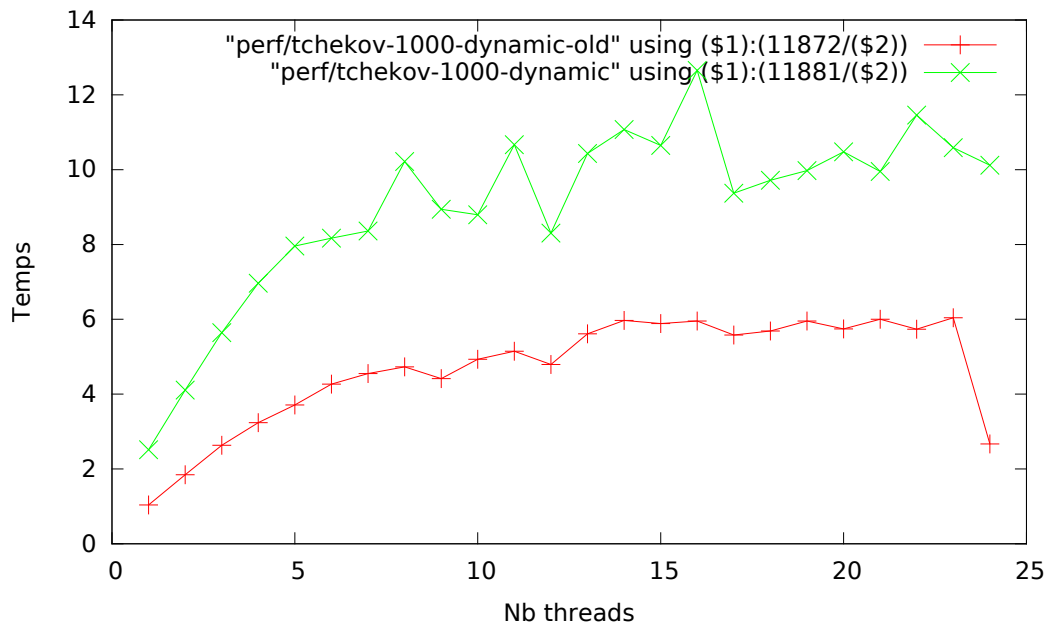


FIGURE 8 – Expérience sur Tchekov, impact de la politique dynamique et du tri

Stratégie de distribution guided

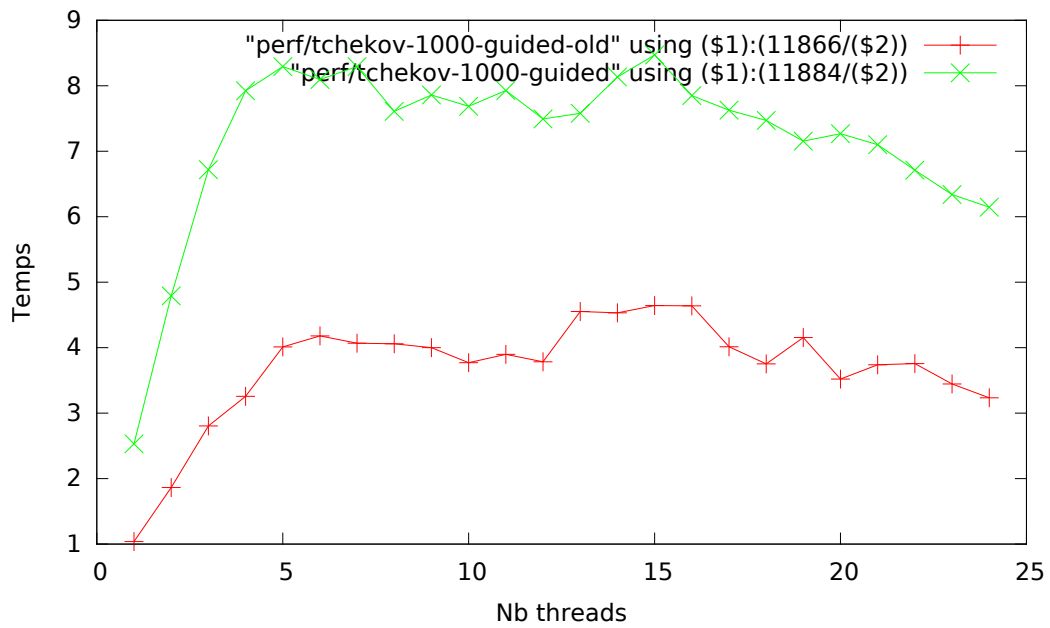


FIGURE 9 – Expérience sur Tchekov, impact de la politique guided et du tri

Comparaison des différentes stratégies

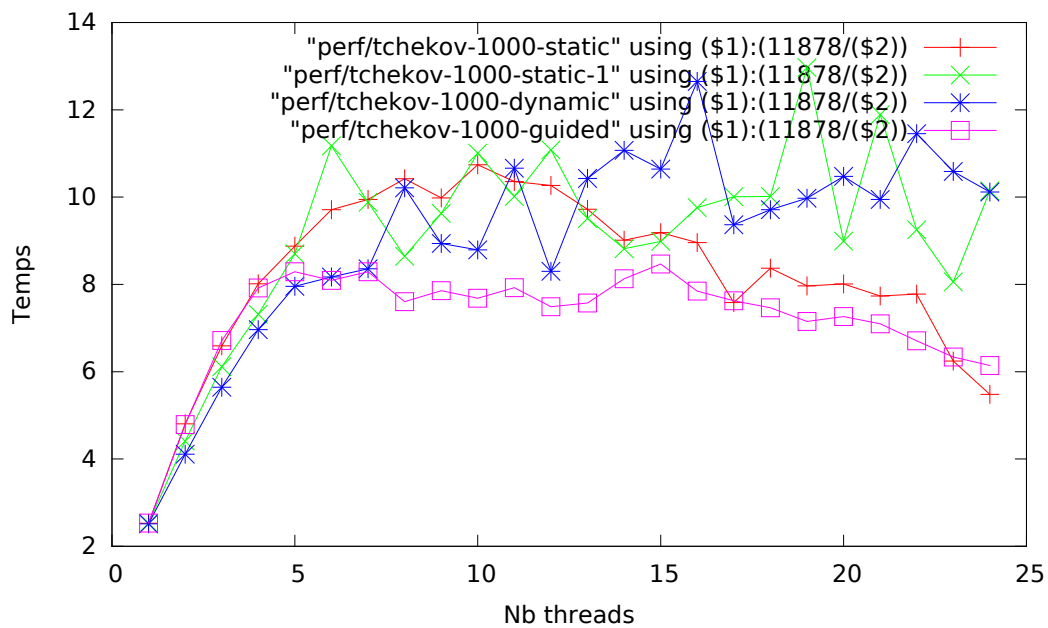


FIGURE 10 – Comparaison stratégies avec tri

Commentaire

Peu importe la stratégie, la version triée présente une meilleure accélération. Le calcul de la force (fonction *omp_force*) étant différente pour chaque atome, les politiques de distribution cyclique et dynamique d'indices affichent un bon speedup. Ceci s'explique par le fait qu'un atome peut avoir plus de voisins qu'un autre (après tri), le problème n'est donc plus régulier.

2.4 Variation sur le domaine initial

Les courbes

Cette expérience a été réalisée avec 1000 atomes pour 100 itérations

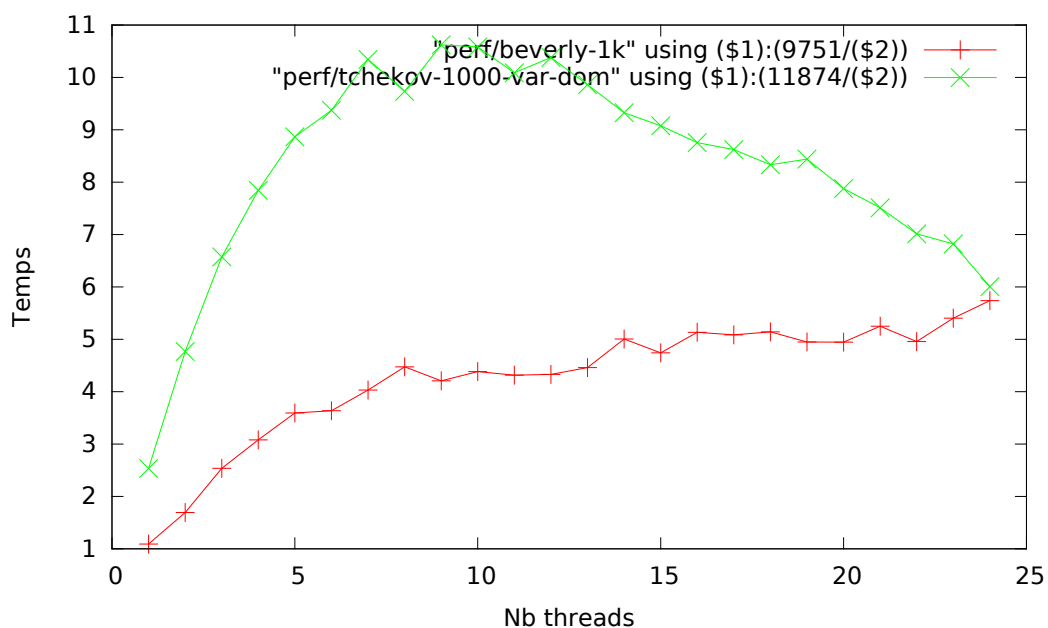


FIGURE 11 – Expérience sur Tchekov, impact du tri

Cette expérience a été réalisée avec les fichiers de configuration choc2

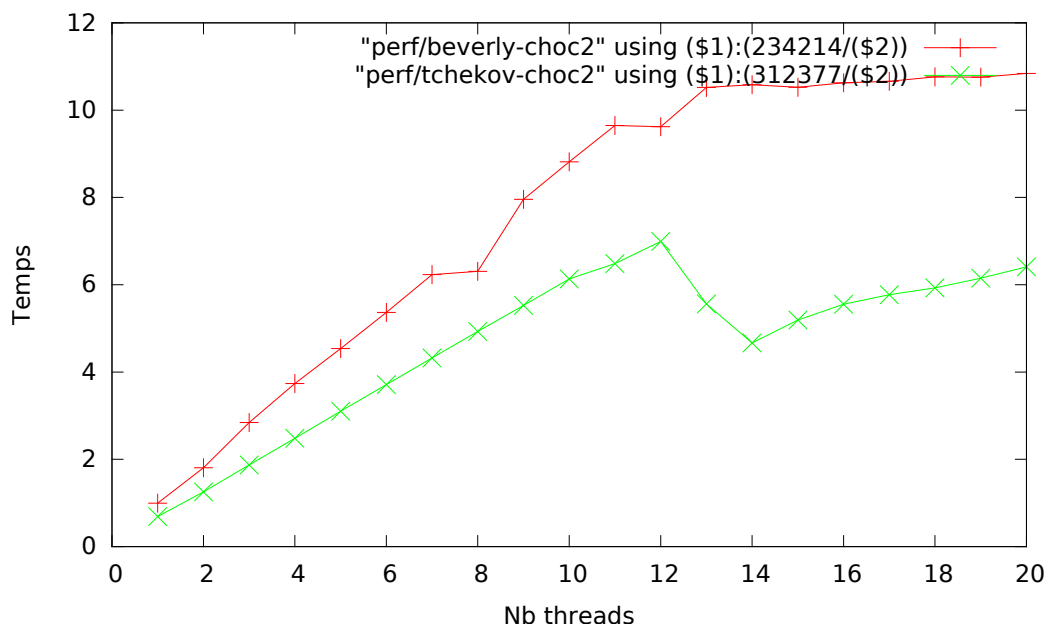


FIGURE 12 – Expérience sur Tchekov, impact du tri

Cette expérience a été réalisée avec les fichiers de configuration choc4

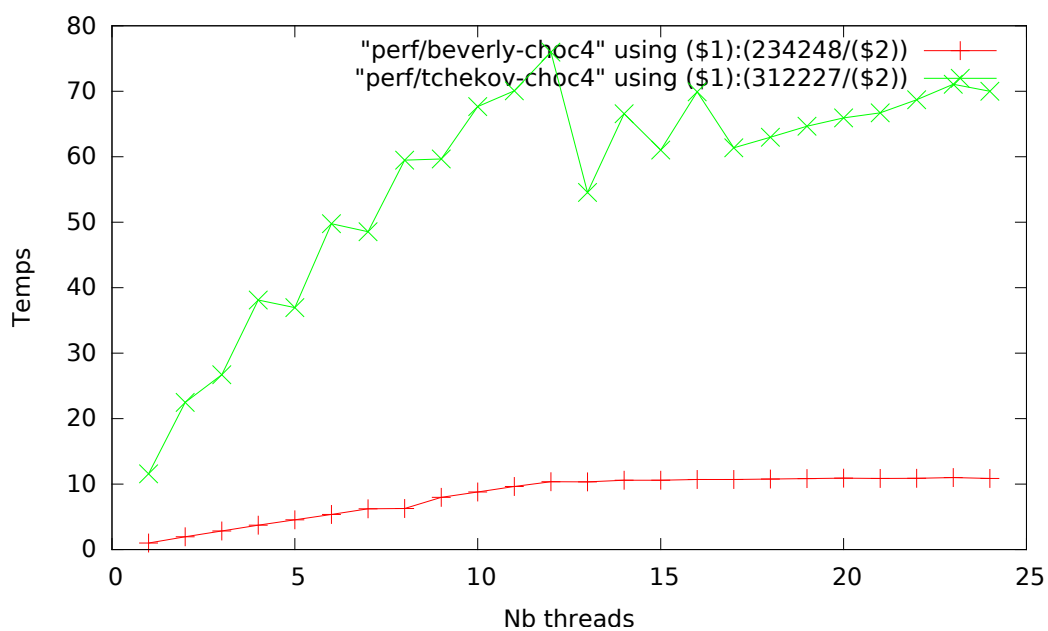


FIGURE 13 – Expérience sur Tchekov, impact du tri

Commentaire

Avec 1000 atomes et le fichiers de configuration choc4, l'impact du tri est encore remarquable, cependant, avec le fichier de configuration choc2, la version non triée est beaucoup plus performante. Cela s'expliquerait par le fait que la plupart des atomes possède la même coordonnée en z (le tri n'a aucun effet). On atteint donc ici, les limites du tri des atomes suivant l'axe z.

3 Tri par boîte

Cette section présente l'impact du tri par boîte des atomes sur la performance de la simulation.

3.1 Type de machine

Cette expérience a été réalisée sur la machine Tchekov (salle 203) et sur le serveur Boursouf. le nombre d'atomes utilisé est 200 pour 100 iterations.

Les courbes

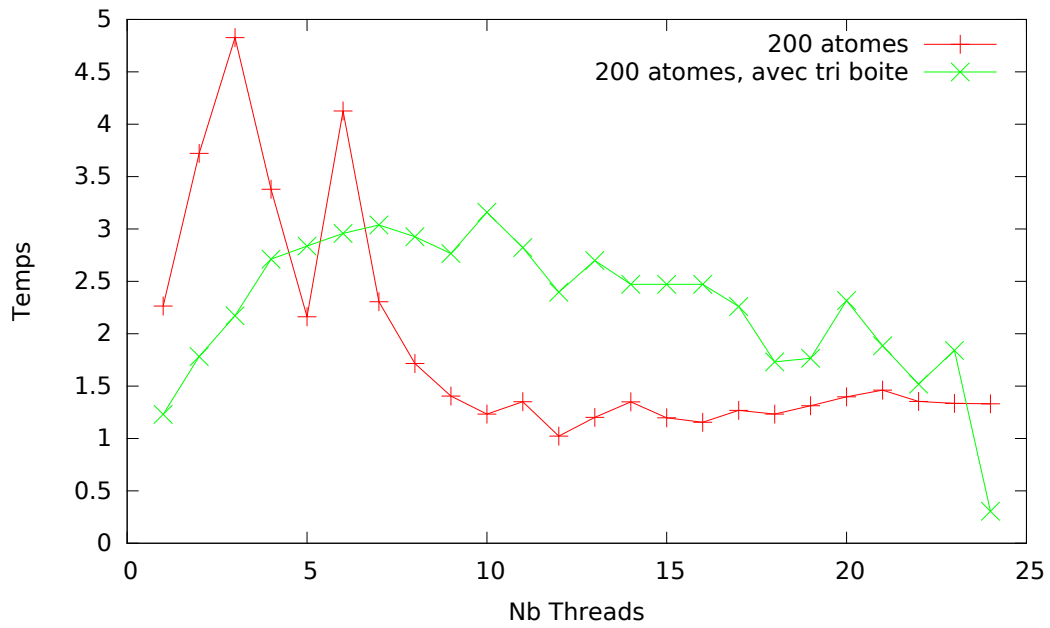


FIGURE 14 – Expérience salle 203, impact du tri par boîte avec 200 atomes

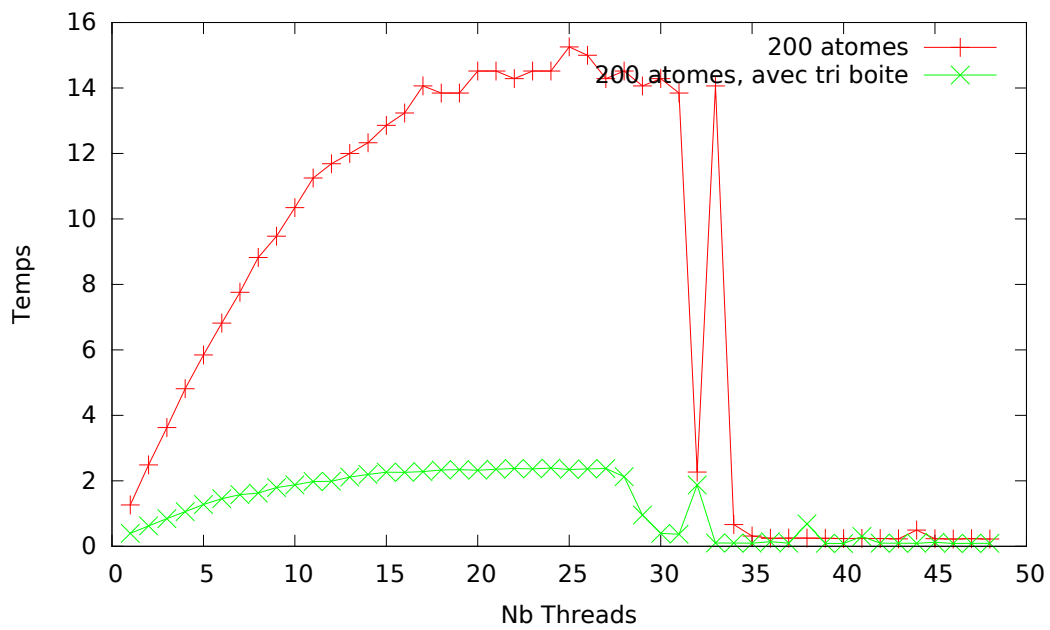


FIGURE 15 – Expérience sur Boursouf, impact du tri par boîte avec 200 atomes

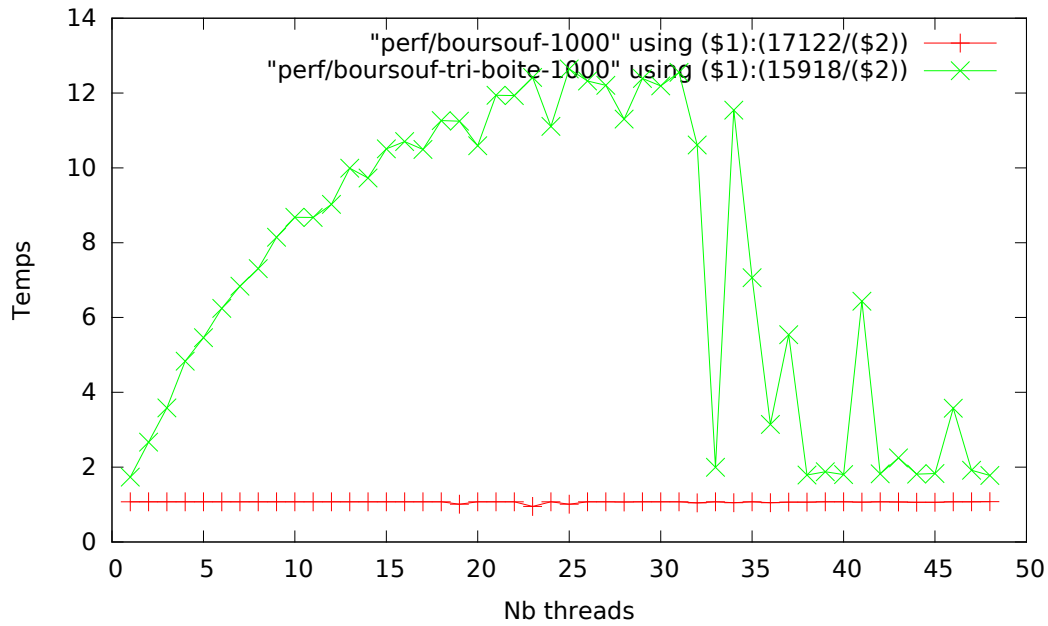


FIGURE 16 – Expérience sur Boursouf, impact du tri par boîte avec 1000 atomes

Commentaire

Comme pour le tri suivant l'axe z , sur la machine Tchekov, le tri a réduit considérablement la charge des threads par unité d'itération, l'accélération est très nette. Cependant, sur le serveur Boursouf, on perd en performance pour une simulation avec peu d'atomes.

Nous supposons que, comme il y a beaucoup de threads et peu de travail, on atteint rapidement l'accélération maximum (bornée par le temps de la version séquentielle d'après la loi d'Amdahl). Cette hypothèse est confirmée par la figure précédente qui représente la comparaison des simulations avec 1000 atomes sur Boursouf avec et sans implémentation du tri par boîte. Ici, la version avec le tri par boîte est beaucoup plus rapide. La version sans tri, elle, atteint rapidement ses limites (manque de cache et false sharing).

Cette observation est aussi valable pour le tri sur l'axe z (cf figure 2).

3.2 Placement des threads et de la mémoire

Il s'agit, ici, d'observer l'influence du placement des threads sur la mémoire en utilisant la variable *affinity*. Cette expérience a été réalisée sur la machine Tchekov (salle 203) et sur le serveur Boursouf avec 200 atomes pour 100 iterations.

Les courbes

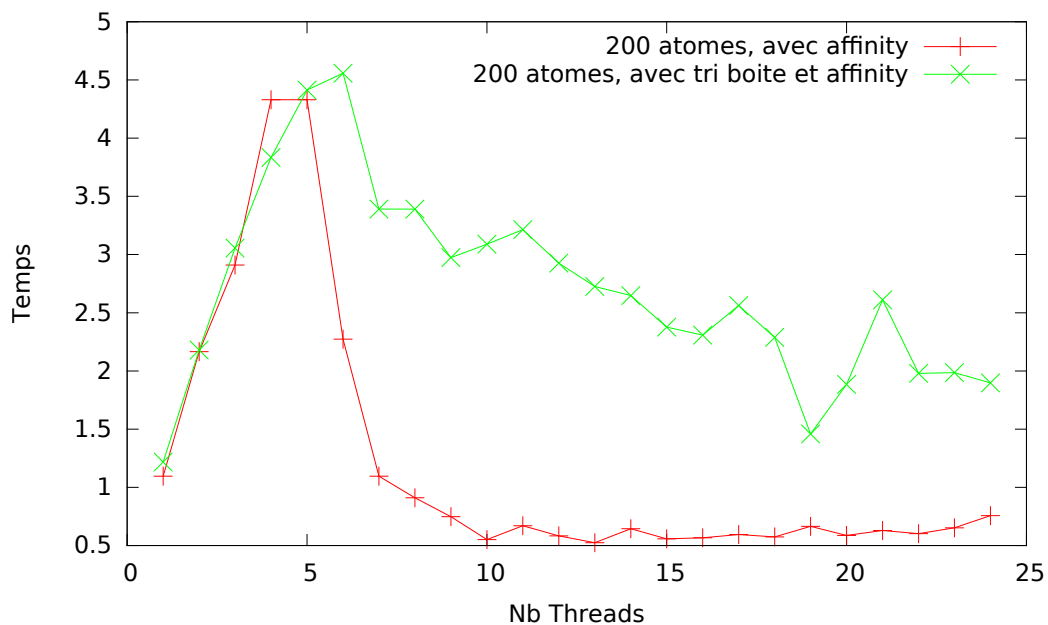


FIGURE 17 – Expérience salle 203, impact de l'affinité et du tri par boîte

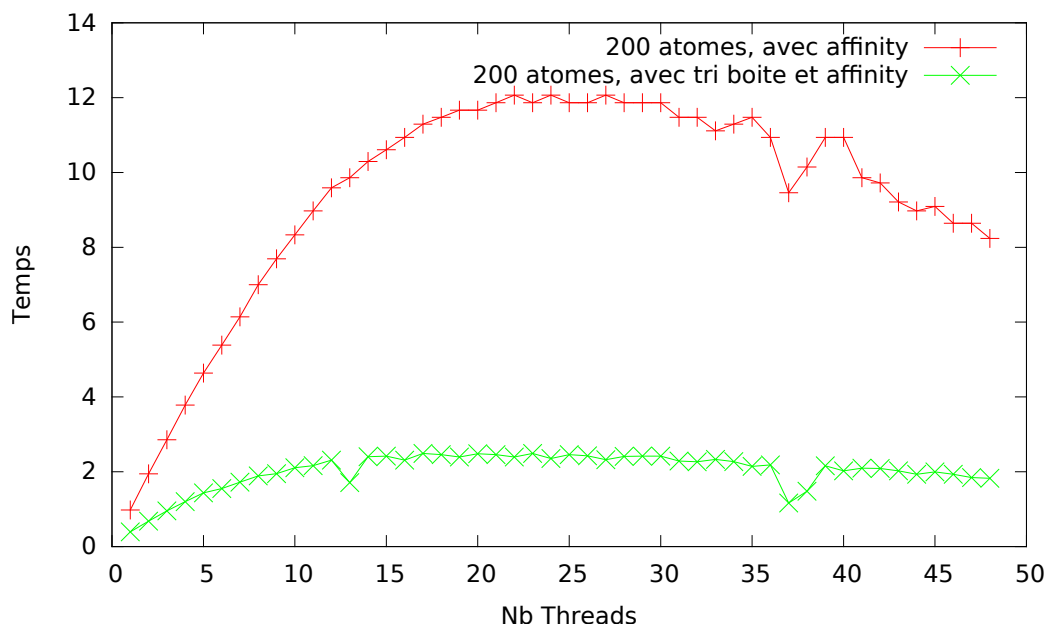


FIGURE 18 – Expérience sur Boursouf, impact de l'affinité et du tri par boîte

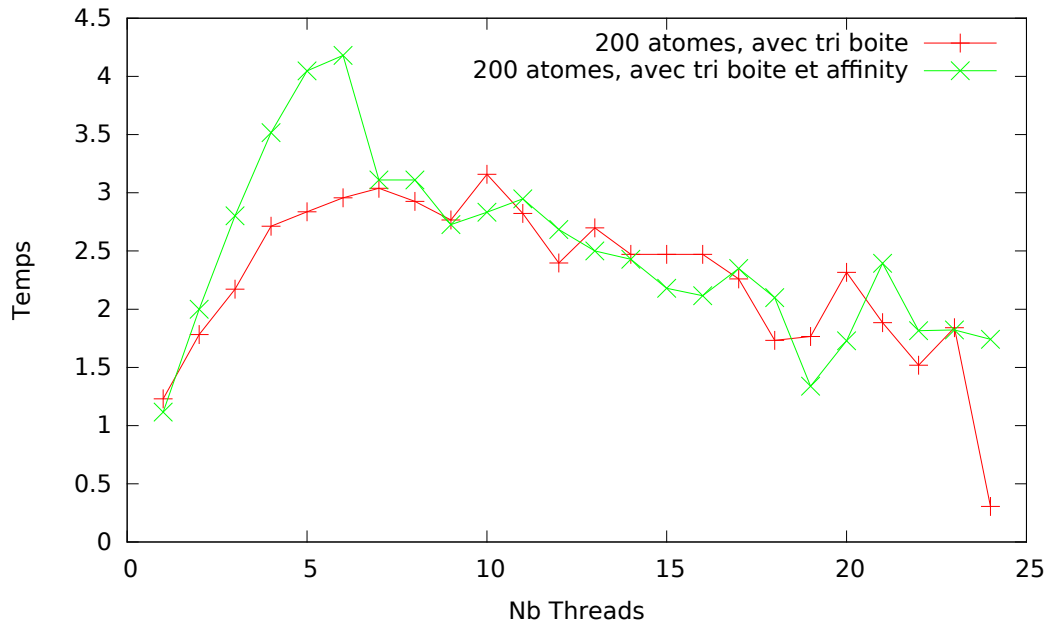


FIGURE 19 – Expérience salle 203, impact de l’ajout de l’affinité sur le tri par boîte

Commentaire

Ces observations concordent aussi avec les résultats des comparaisons de l’affinité avec le tri sur z (voir commentaires de la partie 2.2).

Sur Tchekov, on constate que l’affinité n’a aucun effet sur les performances, comme le prouvent les figures 15 et 19 qui sont très similaires. Ceci est aussi valable avec la simulation triée sur z (cf figure 5).

3.3 Politique de distribution d’indice

Les expérimentations suivantes ont été réalisées avec 1000 atomes et 100 itérations.

Stratégie de distribution statique

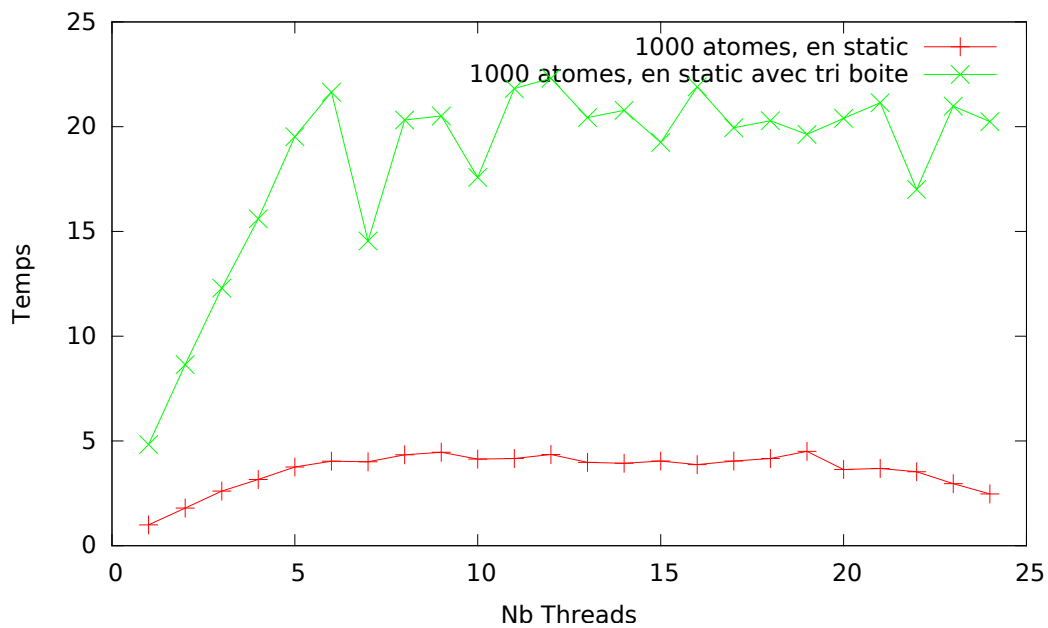


FIGURE 20 – Expérience sur Tchekov, impact de la politique statique et du tri

Stratégie de distribution cyclique

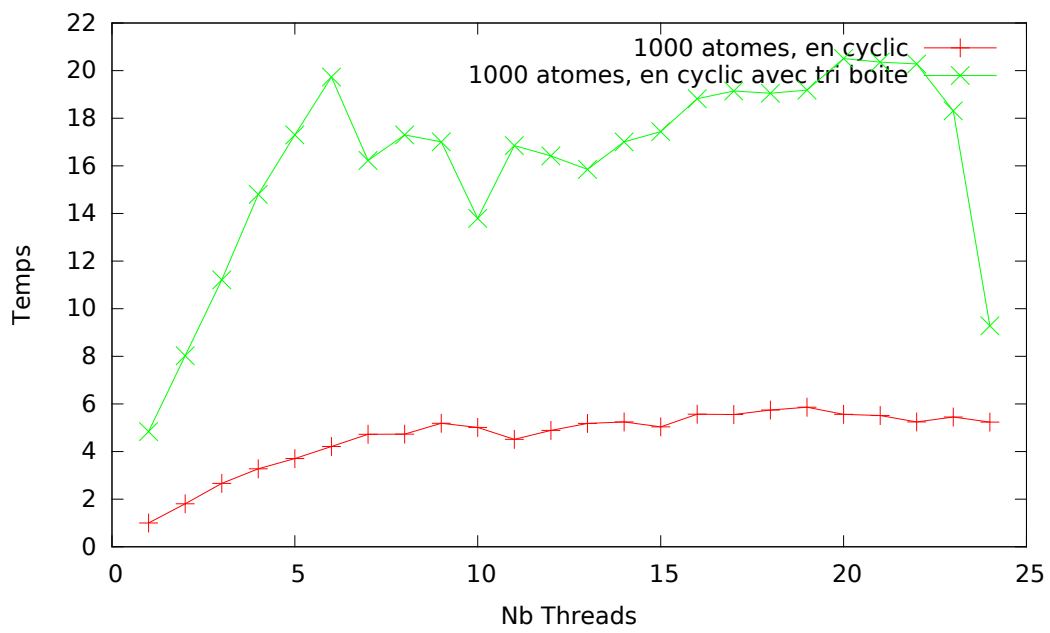


FIGURE 21 – Expérience sur Tchekov , impact de la politique cyclique et du tri

Stratégie de distribution dynamique

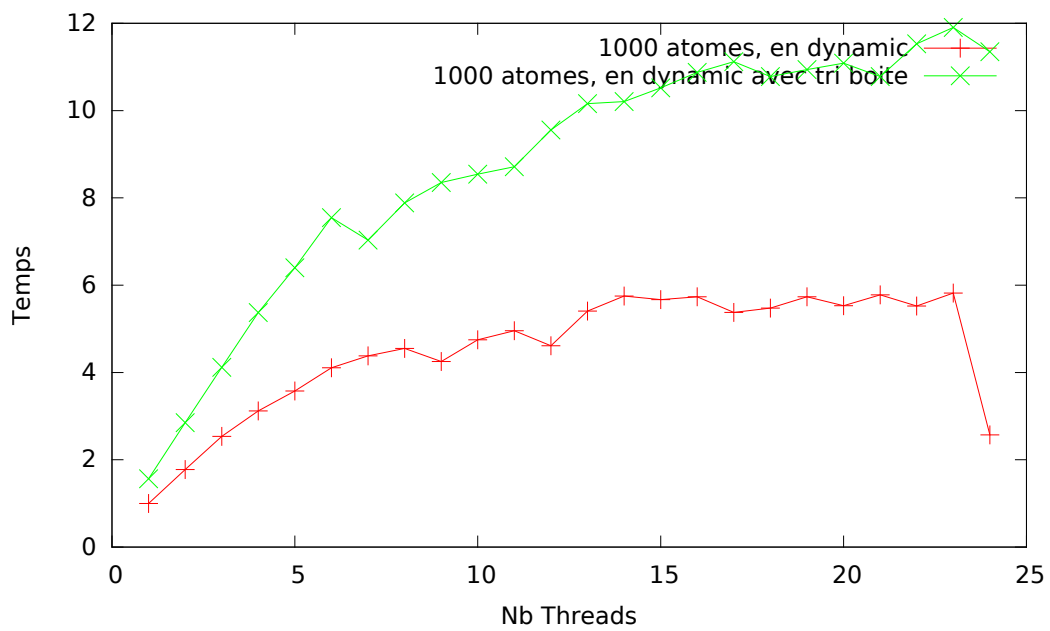


FIGURE 22 – Expérience sur Tchekov, impact de la politique dynamique et du tri

Stratégie de distribution guided

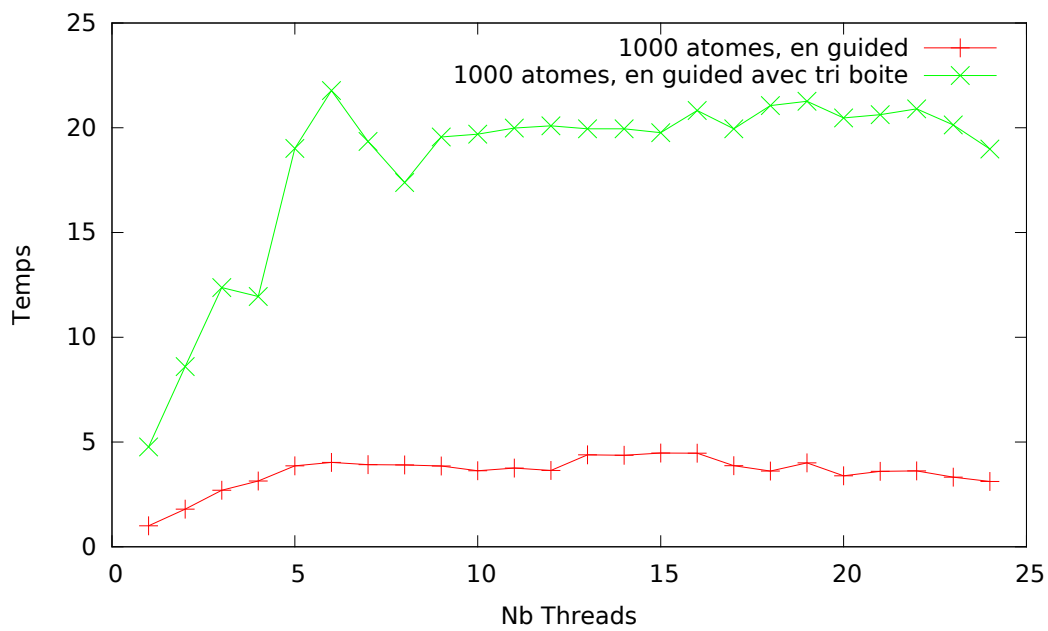


FIGURE 23 – Expérience sur Tchekov, impact de la politique guided et du tri

Comparaison des différentes stratégies

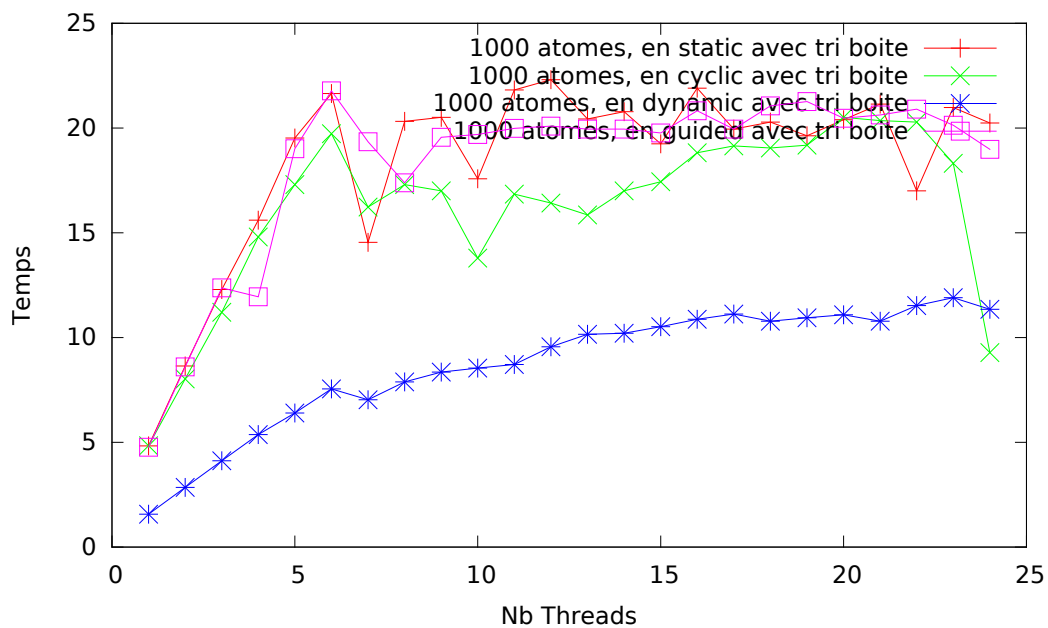


FIGURE 24 – Comparaison stratégies avec tri

Commentaire

Peu importe la stratégie, la version triée présente une meilleure accélération. Ce sont ici les politiques de distribution statique, cyclique et guided d'indices qui affichent le meilleur speedup. Cette observation nous semble étrange, le nombre d'atomes dans une boîte pouvant varier suite à chaque tri, le problème n'est pas régulier.

3.4 Variation sur le domaine initial

Les courbes

Cette expérience a été réalisée avec 1000 atomes pour 100 itérations

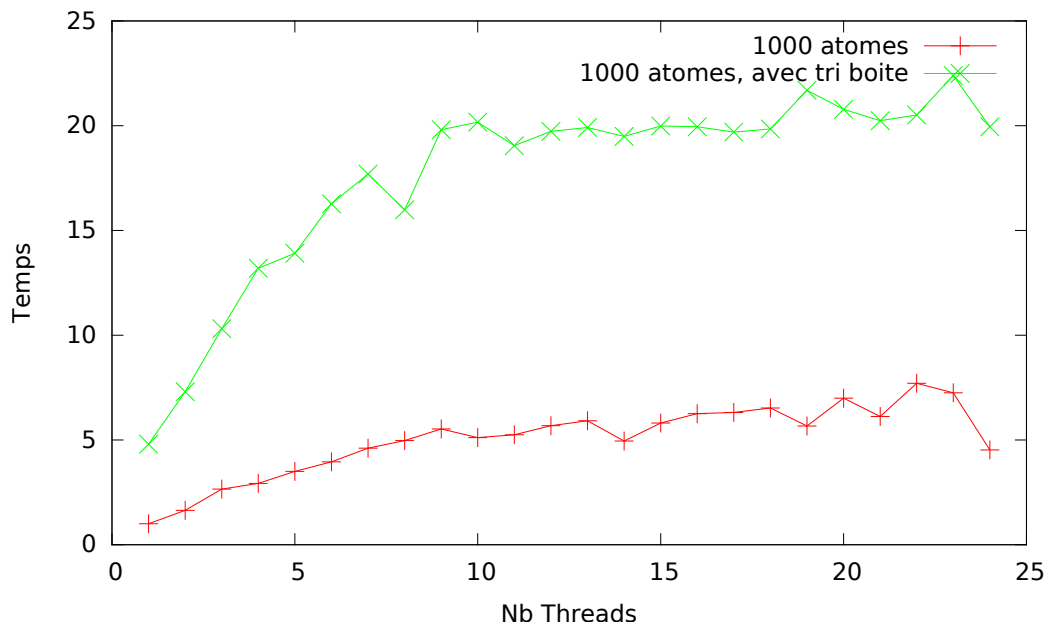


FIGURE 25 – Expérience sur Tchekov, impact du tri avec un domaine initial de 1000 atomes

Cette expérience a été réalisée avec les fichiers de configuration choc2

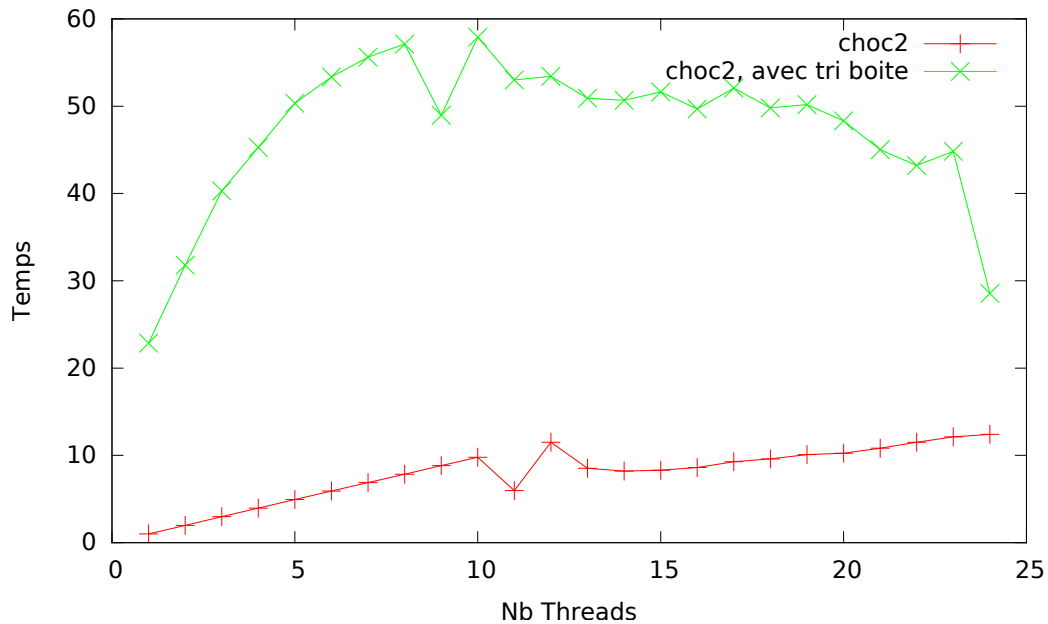


FIGURE 26 – Expérience sur Tchekov, impact du tri avec la configuration choc2

Cette expérience a été réalisée avec les fichiers de configuration choc4

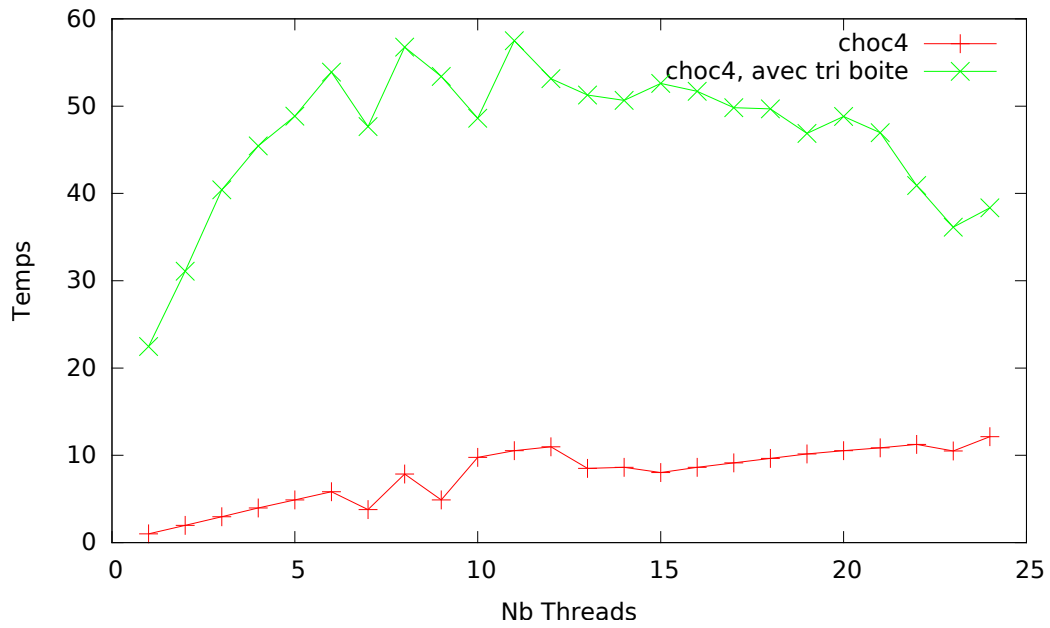


FIGURE 27 – Expérience sur Tchekov, impact du tri avec la configuration choc4

Commentaire

Les trois variations de domaines, départ avec 1000 atomes ou avec les fichiers de configuration choc2 et choc4, affichent un speedup très important par rapport à la simulation non triée. Cela est dû à la baisse de complexité obtenue grâce au tri par boîte. Au lieu de parcourir n atomes pour chaque atomes ($\mathcal{O}(n^2)$), on ne parcourt qu'un nombre k d'atomes bien plus faible que le nombre total d'atomes présents dans la simulation ($\mathcal{O}(n \times k)$).

Conclusion

Le tri suivant l'axe z a permis d'améliorer les performances pour la plupart des expérimentations effectuées. Cependant, l'implémentation du tri par boîte nous a donné de meilleurs résultats pour les simulations, plus particulièrement, pour les simulations comprenant de nombreux atomes, comme pour les départ avec un domaine de 1000 atomes ou avec les fichiers de configuration choc2 ou choc4.