

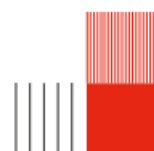
Optimisation et parallelisation OpenMP d'addition et produit de deux matrices denses

Rapport de Bureau d'étude

Luc-Christelle Nguyen et Alicia Perrin

Institut National des Sciences Appliquées de Toulouse

24 octobre 2025



Résumé

Dans ce rapport, nous allons tester plusieurs méthodes différentes afin d'optimiser des calculs matriciels (somme, produit scalaire, produit matricielle). A COMPLETER

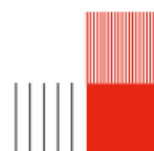
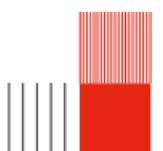


Table des matières

1	Modifier l'accès à la mémoire pour additionner deux matrices	2
2	Compilation reliant les bibliothèques OpenMP et BLAS	3
3	Parallelisation OpenMP CHANGER	4
4	Nombre de threads utilisé pour BLAS3	5
5	Utiliser les blocs du cache	6



Chapitre1

Modifier l'accès à la mémoire pour additionner deux matrices

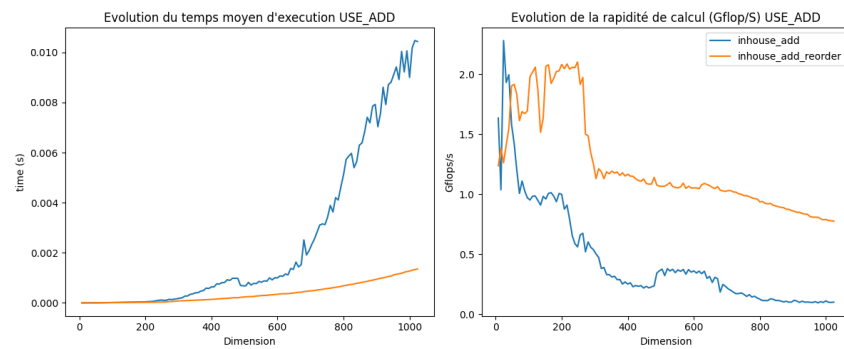
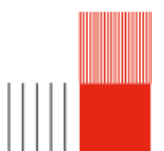


FIGURE 1.1 – Différences de performances en fonction de l'ordre d'accès à la mémoire

Modification de l'ordre d'accès à la mémoire : On lit d'abord les lignes pour une meilleure utilisation du cache. L'efficacité est doublée en moyenne.



Chapitre2

Compilation reliant les librairies OpenMP et BLAS

ECRIRE EN FRANCAIS CE QUE FONT LES BLAS

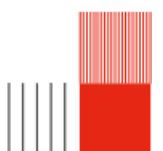


FIGURE 2.1 – Performances BLAS 1, 2, 3 sans optimisation Openblas



FIGURE 2.2 – Performances BLAS 1, 2, 3 avec optimisation Openblas

On peut voir une nette amélioration des performances lorsque on utilise la librairie openblas qui ...
mettre ce qu'elle fait Pour BLAS3 5Gflops -> 400Gflops.



Chapitre3

Parallelisation OpenMP CHANGER

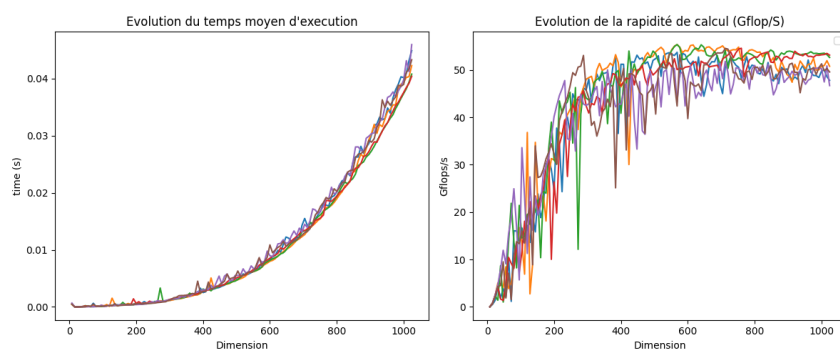


FIGURE 3.1 – Trouver titre

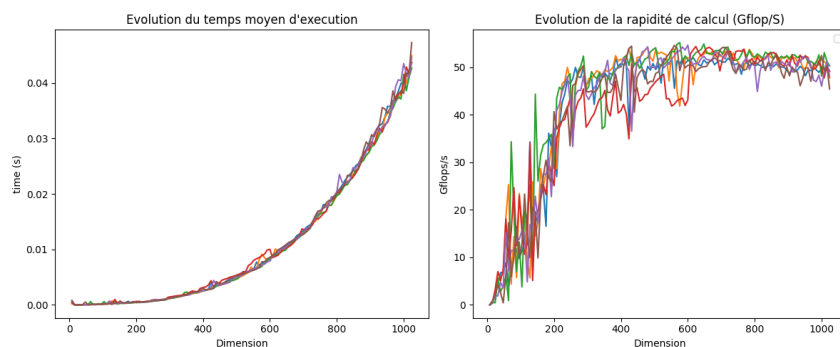
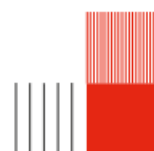


FIGURE 3.2 – Trouver titre

Comparer les option de parallelisation Static et Dynamique avec des nombres de Thread différents. On peut remarquer que cchanger ses options n'a pas une réelle influence.



Chapitre4

Nombre de threads utilisé pour BLAS3



FIGURE 4.1 – Trouver titre

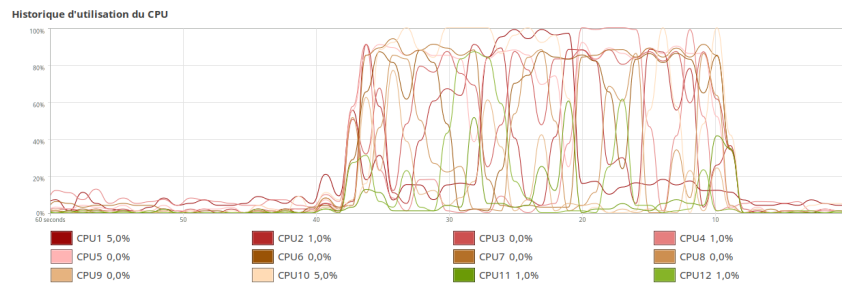
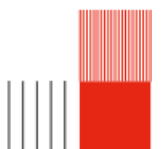


FIGURE 4.2 – Trouver titre

Fait des tests de 1 à 12. Le nombre optimal c'est 6 parce que plus on parrallelise, plus on doit partager des données entre les différents threads.

Expliquer que 6 threads qui ravail en permanence mais ce ne sont pas tout le temps les même.



Chapitre5

Utiliser les blocs du cache

Expliquer la technique du block.

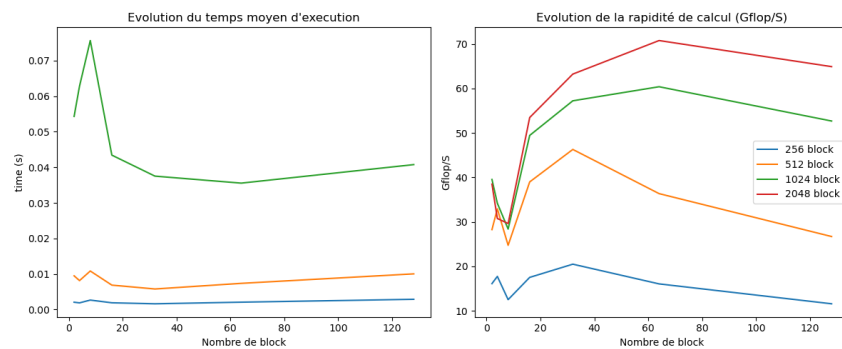


FIGURE 5.1 – Trouver titre

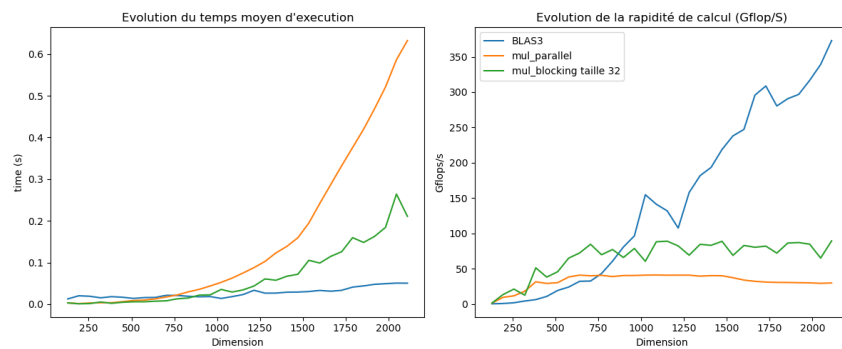
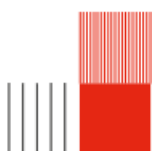


FIGURE 5.2 – Trouver titre

On a commencé par une recherche de la taille de block optimale en fonction de différentes tailles de matrices. On a choisi 32 (mieux que 64).

Puis on a comparé cette technique avec openblas et la parrallelisation.

On peut observer que la librairie openblas est la meilleure solution pour optimiser les performances de calcul en partageant efficacement les thread.



Bibliographie

