

Mini Projet CISC 2022-2023 (AVANT 31/01/23 minuit)

Profils de grandeurs physiques, échantillonnage en tranches

Objectif :

Créer un outil d'analyse in-situ qui extrait un profil (une courbe) d'une grandeur physique (énergie cinétique, force, etc.) selon une direction donnée.

Pour construire ce profil selon une direction, on va échantillonner notre domaine de calcul en le décomposant en "tranches" perpendiculaires à la direction souhaitée.

Contexte :

Comme pour les TPs vous allez créer un nouvel objet dans le code de dynamique moléculaire exaStamp. Dans ce code les particules (atomes) sont stockées dans des cellules. Ces cellules composent elles-mêmes une grille régulière. Un morceau de cette grille (sous-domaine) est stockée sur chaque processus MPI. Attention au découpage en sous domaine, ce sont des « boîtes » rectangulaires, mais le découpage peut être quelconque.

Attendus :

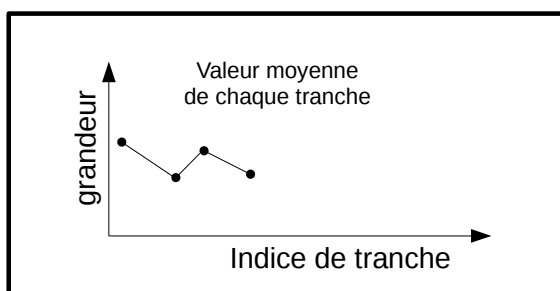
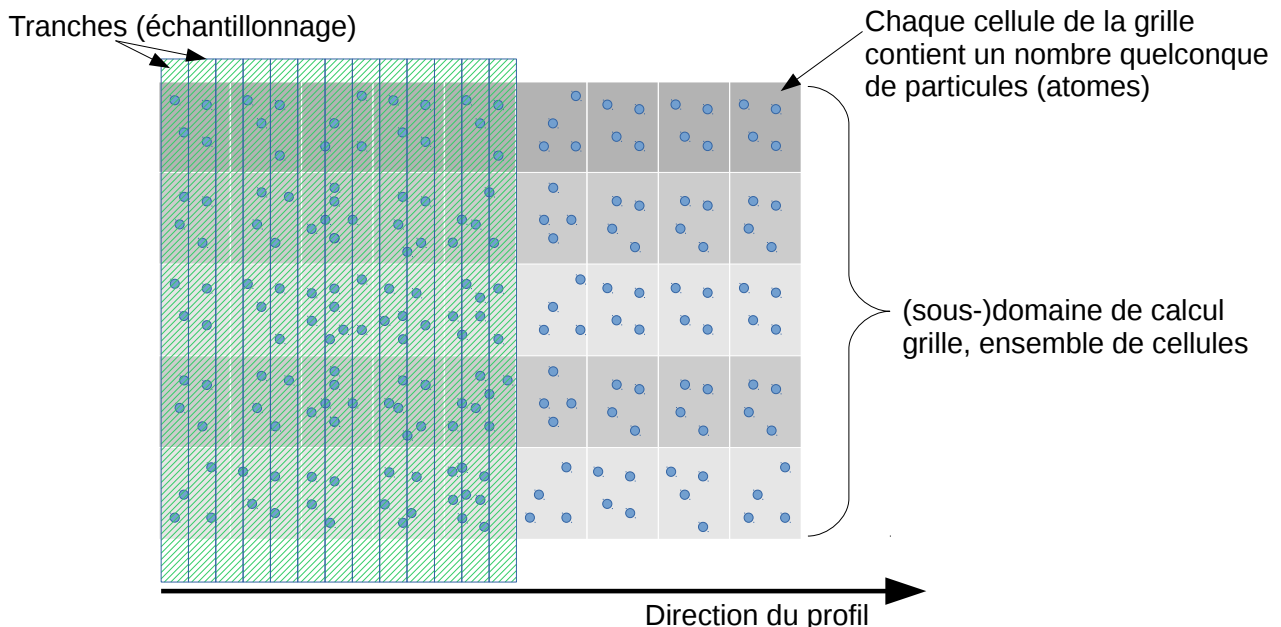
Vous allez partir de votre branche **enseirb- $\{USER\}$** (celle que vous avez créé au début des Tps).

Le fichier à modifier est **src/tutorial/slice_plot.cpp** (rien ne vous empêche d'en créer d'autres).

Un fichier d'entrée d'exemple est également fourni : **data/samples/tutorial_slice_plot.msp**

Vous réaliserez l'outil d'analyse qui produit un fichier de courbe à partir des paramètres suivants :

- Point de départ des tranches (paramètre **origin**, de type Vec3d)
- direction perpendiculaire aux tranches (paramètre **direction**, de type Vec3d)
- espacement (=épaisseur) des tranches (paramètre **thickness**, de type double)
- nombre de tranches (paramètre **nslices**, de type long). Si **nslices** vaut 0, alors le nombre de tranches est calculé automatiquement pour couvrir toutes les particules au-delà du plan de référence.



Résultat attendu :
la courbe dans un fichier bi-colones

Attendus :

vous devez rédiger un rapport d'un maximum de 4 pages décrivant vos choix de conception, comment vous avez géré les parallélismes MPI et OpenMP. Ce rapport devra également contenir une étude performance du parallélisme OpenMP (courbe d'accélération, analyse qualitative, etc.). La conclusion et la perspective du rapport décriront ce que vous auriez aimé développer si vous aviez eu plus de temps.

En plus du rapport au **format PDF**, vous devrez fournir un *bundle* git de vos sources afin de vérifier que votre code s'exécute et fourni le bon résultat.

Construire le *bundle* git avec la commande suivante (dans votre répertoire de source) :

```
git commit -am « commentaires de commit »
```

```
git bundle create ~/nom-prenom.bundle master~20..enseirb- $\$$ {USER}
```

Ce mini-projet peut se faire seul ou en binôme. La notation sera adaptée pour les binômes.

Les points importants pour la notation :

Au minimum :

Code qui fonctionne avec parallélisme MPI et OpenMP, avec direction de plan quelconque.

Rapport PDF avec explication de l'implémentation et détails de l'algorithme de parallélisation utilisé, son principe de fonctionnement, sa complexité théorique (paramètres influant sur sont temps d'exécution).

Analyse de performance, avancer des explications (en les argumentant) de ces performances.

Propositions d'optimisations et/ou d'algorithme différent que l'on aurait pu utiliser.

Pour avoir des points supplémentaires :

Gestion correcte du déplacement des particules (max_displ)

Gestion du paramètre nslices=0 : calcul automatique du nombre de tranches pour atteindre la particule la plus lointaine (distance orientée max) du plan de référence.

Qualité du rapport.