

Optimisation d'une simulation d'interactions gravitationnelles à N Corps

ING2-GSI – Projet Agile

Année 2017–2018



1 Introduction

Une simulation du “*problème à N corps*” consiste à déterminer par le calcul informatique le mouvement de N masses sous l'effet des forces d'attraction gravitationnelles entre elles [1, 2]. En effet, il est possible de calculer la position et vitesse de plusieurs corps en utilisant les équations classiques de Newton. Un exemple simplifié des équations régissant ce type de systèmes peut s'exprimer comme suit [3]:

```
DO t = 1,N t
  DO i = 1,N x
    force = calculate_force(position[x1[i]], position[x2[i]])
    f[x1[i]] = f[x1[i]] + force
    f[x2[i]] = f[x2[i]] - force
  END DO
  position[1:Nf] = update_position(f)
  velocity[1:Nf] = update_velocity(f)
END DO
```

Du point de vue computationnelle, pour chaque corps il faudra calculer les forces exercées par les autres corps, ce qui devient un algorithme de complexité $O(N^2)$. Heureusement, il existe des simplifications qui permettent de réduire la complexité afin de traiter un problème dans une période de temps limitée. Malgré tout, la simulation d'un problème de N corps, quand N devient grand, requiert l'utilisation d'infrastructures de calcul parallèle.

2 Objectif et méthodologie

L'objectif de ce projet est le développement d'un calcul efficace pour connaître à chaque instant la position et vitesse de N corps qui interagissent entre eux. Par *efficacité* entendons un programme parallèle capable de s'adapter à l'architecture matérielle sur laquelle il tourne selon la taille du problème. Le déroulement du projet se fera selon la méthodologie Agile.

3 Organisation et planification des User Stories au sens AGILE

3.1 Themes & User Stories AGILE

Objectif

J'ai besoin de connaître la position et vitesse de N corps dans un instant t_i à partir de leurs positions et vitesses initiales.

Thème récupération des données

US1 : J'ai besoin d'un ou plusieurs jeux de données réelles.

Thème simulation

US2 : J'ai besoin de calculer la position et vitesse de chaque corps à partir de sa trajectoire et de l'influence des corps voisins.

US3 : J'ai besoin d'optimiser le temps d'exécution selon la taille du problème.

US4 : J'ai besoin d'optimiser le temps d'exécution selon le type d'architecture parallèle.

Thème affichage

US5 : J'ai besoin d'afficher les résultats de façon périodique.

3.2 Planification

1 run Agile par semaine sur 7 semaines :

Sprint 0 : Run Agile d'échauffement pour définition des données et prise de connaissance des machines parallèles disponibles.

Sprint 1 : US1 - Obtention d'un ou plusieurs jeux de données.

Sprint 2 : US2 - Calcul de la position de N corps.

Sprint 3 : US3 - Optimisation selon la taille du problème.

Sprint 4 : US4 - Optimisation selon l'architecture parallèle de la machine.

Sprint 5 : US4 - Optimisation selon l'architecture parallèle de la machine.

Sprint 6 : US5 - Afficher les résultats.

3.3 Répartition des tâches techniques

Les différentes User Stories seront décomposées en tâches techniques. Par exemple :

Calcul de la position de chaque corps

Pour chaque corps

- Calculer sa position et vitesse à partir de sa position précédente et des forces exercées par les autres corps.
- Mettre à jour les nouvelles positions et vitesses dans les matrices correspondantes.

Optimisation selon l'architecture parallèle de la machine

Pour chaque simulation

- Décider le type de parallélisation à faire (OpenMP, MPI, hybride) selon la taille et type de nœud ou nœuds de calcul utilisés.
- Évaluer et comparer la performance.

4 Ressources

N'étant pas physiciens mais ingénieurs, nous pouvons partir d'un code simple de simulation de N Corps, comme par exemple celui de [4]. Des exemples de jeux de données peuvent se trouver dans les exemples d'interactions de particules des logiciels de simulation moléculaire comme LAMMPS [5].

Le superordinateur de calcul parallèle *FinisTerra II* du CESGA sera disponible pendant toute la durée du projet. Vous pouvez accéder avec vos comptes d'utilisateur. Regardez <https://arel.eisti.fr/documents/56269> pour plus d'information sur l'accès et l'utilisation de l'infrastructure.

5 Modalités

Ce projet est à effectuer par groupe de 3, la composition des groupes étant libre. Les choix algorithmiques et d'implémentation sont libres. Vous devez mettre en œuvre vos connaissances de programmation système, parallèle et réseau. Tout le travail doit être effectué en utilisant le GitLab de l'école (codes sources, rapport, fichiers de tests, etc.).

6 Dates importantes

- **Début du projet** : semaine du 5 février.
- **Fin du projet** : semaine du 30 mars.
- **Soutenance et rendu du dossier de synthèse** : à définir.

7 Notation

Seront notés les points suivants :

- Aptitude Agile (donnée par le coach).
- Notation / appréciation par le client, qui prendra en compte le pourcentage de finition et la qualité. Cela comprendra la compréhension du problème, les techniques et méthodes d'implémentation et, bien entendu, la réponse apportée à l'objectif attendu.
- Une notation du code.
- Une évaluation de l'oral final.
- Une notation du rapport de synthèse.

Références

- [1] *Wikipedia : Problème à N corps*, dernière visite : 18/01/2018.
https://fr.wikipedia.org/wiki/Probleme_a_N_corps.
- [2] *Pourquoi Comment Combien : Le problème à N corps*, dernière visite : 18/01/2018.
<https://www.drgoulu.com/2008/11/16/le-probleme-a-n-corps>.
- [3] J. A. Lorenzo, J. L. Albin, T. F. Pena, F. F. Rivera, and D. E. Singh, "An inspector/executor based strategy to efficiently parallelize n-body simulation programs on shared memory systems," in *Parallel and Distributed Computing, 2007. ISPD '07. Sixth International Symposium on*, pp. 9–9, July 2007.
- [4] N. J. Institute for Advanced Study, *A Starter Code for N-body Simulations*, dernière visite : 18/01/2018. <https://www.ids.ias.edu/piet/act/comp/algorithms/starter>.
- [5] S. Plimpton, *Large-scale Atomic/Molecular Massively Parallel Simulator*, dernière visite : 18/01/2018. <http://lammmps.sandia.gov/>.