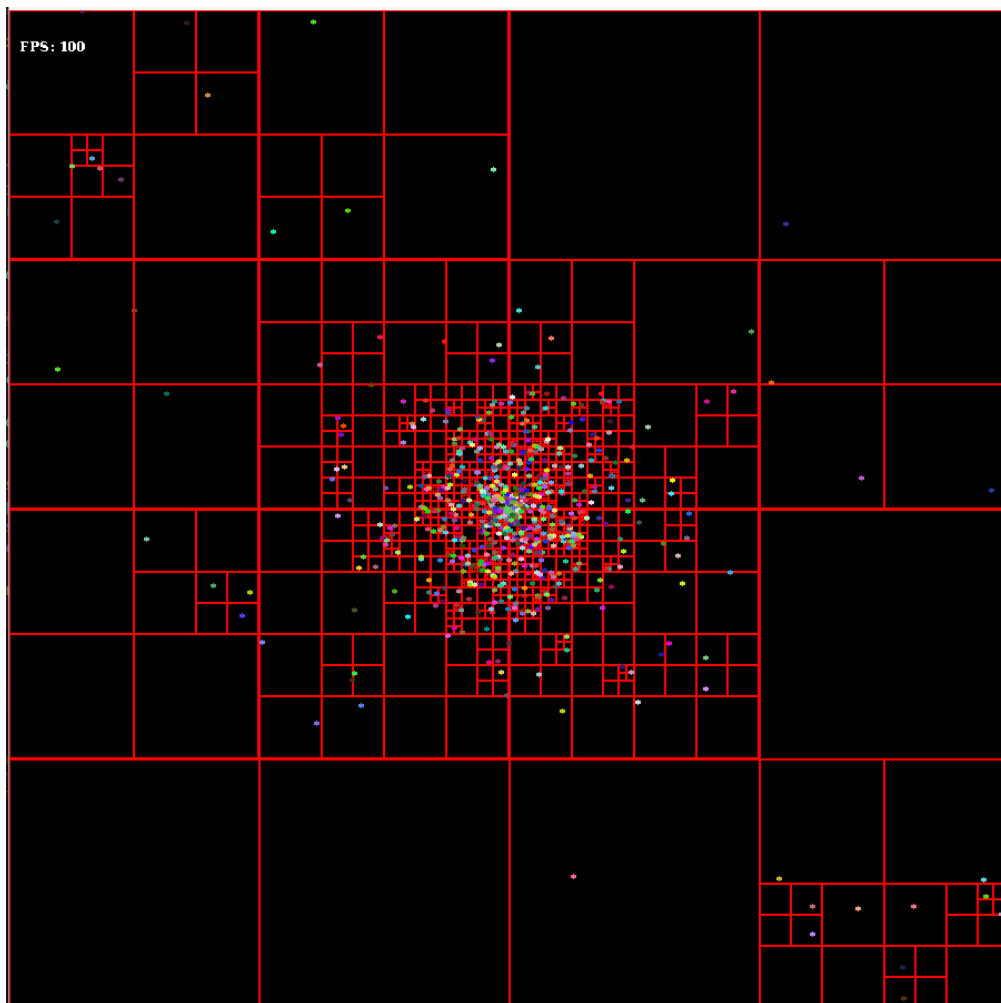


Projet Galaxy Simulation

Rapport de projet



ESIFE INFO 1



Table des matières

Manuel utilisateur	2
Guide compilation :.....	2
Options :	2
Etat final du projet	2
Version A & B:.....	2
Améliorations :	2
Déroulement du projet	3
Organisation :	3
Problèmes rencontrés :	3
Comparaison A vs B	4
Performances :.....	4
Conclusion & complexité cachée :	4

Manuel utilisateur

Guide compilation :

Une fois l'archive « BADAROU_COMMUNIAU_Galaxy_Simulation.zip » extraite, allez dans le dossier du même nom. Pour compiler un *make* suffit. Un *make clean* et *make re* est aussi disponible si besoin de nettoyage ou recompilation.

Pour lancer le programme (version A par défaut) :
chemin vers galaxy.out et chemin vers un fichier d'étoile (stars_X.txt).

Depuis le répertoire racine ça donne :

`./bin/galaxy.out ./data/stars_X.txt` (X correspondant au nombre d'étoiles)

Options :

--quad

option pour lancer la version quadtree

--show

option pour afficher le quadrillage de segmentation lors du quadtree (nécessite l'option --quad)

Etat final du projet

Version A & B:

Les deux versions sont fonctionnelles.

La version A performe uniquement sur les fichiers contenant peu de corps, sa complexité est de $O(n^2)$.

La version B basé sur un quadtree, est fonctionnel sur tous les fichiers et est de complexité $O(n \log n)$.

Améliorations :

- Quadrillage : fonctionnel .
- Affichage des FPS : fonctionnel.
- Fusion des corps : mise en place mais problème pour faire disparaître un corps fusionné. La passer à NULL et free ne suffisaient pas à la faire disparaître totalement de l'affichage graphique. Solution trouvée : passer la couleur du corps à noir.

Déroulement du projet

Organisation :

On s'est, dans un premier temps, concerté à propos de ce que l'on voulait faire. Nous avons remanié la formule de calcul de la force et de la vitesse pour n'en faire qu'une.

Ensuite, Maxime s'est occupé de mettre en place la version A tout en pensant à l'aspect dynamique (pouvoir changer la couleur ou la taille des corps en les passant en tant que champs dans les structures, plutôt qu'en dur avec des constantes).

Pour la version B c'est Mikdaam qui s'en est chargé, il a implémenté le quadtree ainsi que la mise en place de certaines options. Il a aussi travaillé sur les options au lancement du programme tel que **--show**.

Problèmes rencontrés :

Lors de la mise en place du quadtree, nous avons fait face à un problème que nous n'étions pas capable d'en situer la cause. Nous avons dû faire appel à notre professeur pour résoudre le problème. Finalement c'était l'endroit où était appelé la fonction récursive de mise à jour des positions des corps qui était responsable de ce dysfonctionnement.

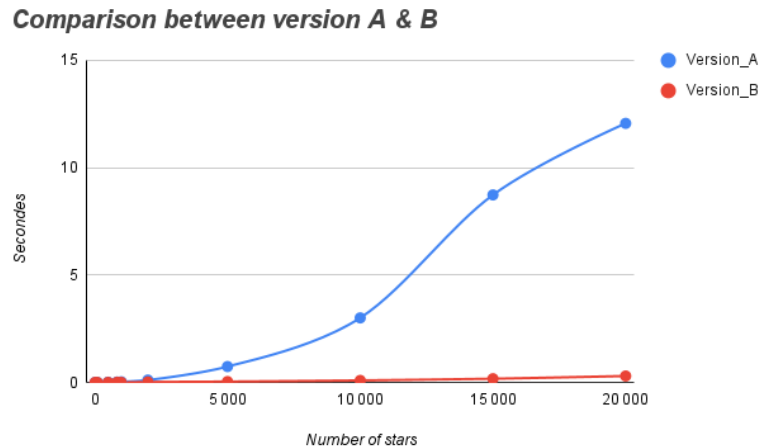
Quant à l'amélioration sur la fusion des corps, comme mentionné plus tôt, même avec un `free()` du corps ainsi qu'une mise à jour de tous ses champs à NULL. Le corps reste visible à l'affichage, d'une taille plus petite que les corps « vivants ».

Comparaison A vs B

Performances :

Nous avons effectué des tests avec des fichiers allant de quelques corps à 20 000 corps.

En voici le graphique représentant le nombre de secondes moyen nécessaire pour effectuer un tour de boucle (soit une mise à jour de toutes les positions des corps).



Pour réaliser ce diagramme nous avons pris une moyenne du temps d'exécution de mise à jour des corps pour chaque fichier.

Conclusion & complexité cachée :

On remarque directement que la version A n'est plus viable passée les 5 000 corps. Par ailleurs, sa complexité « généralisé » est $O(n^2)$ mais il se trouve qu'il y a un parcours $O(n)$ en plus. Sa complexité cachée est donc un $O(n)$ en plus du $O(n^2)$. Soit $O(n^2+n)$.

Dans la version B qui est initialement en $O(n \log n)$, il a aussi deux parcours en $O(n)$ externe à $O(n \log n)$. Sa complexité cachée étant un $O(2n)$ on obtient pour finir avec une complexité de $O(n \log n + 2n)$.

Toutes ces complexités sont visibles depuis leurs fonctions `update_galaxy_simple()` et `update_galaxy_with_quadtree()` respective à chaque version.