Projet PSFPN : le plus grand rectangle

Version du 27 janvier 2017

Ce projet aborde un problème d'algorithmique simple : le calcul du plus grand rectangle à l'intérieur d'un nuage de points (le nombre de points N pouvant atteindre plusieurs millions). Ce problème, qu'on rencontre par exemple dans des concours de programmation (séquentielle), permet en effet d'aborder différents algorithmes, plus ou moins efficaces en séquentiel. Or ces différents algorithmes présentent un « potentiel » de parallélisation très variable.

Le but de ce projet est donc d'implémenter des versions séquentielles puis parallèles de ces algorithmes, sur CPU muli-coeur, afin de les comparer et de déterminer si et comment l'impact du parallélisme modifie le classement (en termes de temps de calcul) de ces divers algorithmes.

1 Présentation du problème

On considère la région rectangulaire du plan déterminée par le rectangle $(0,0)(0,h)(l,h)(l,0)$, h et l étant deux entiers positifs, ainsi que n points à coordonnées entières à l'intérieur de cette région. On souhaite dessiner un rectangle dont la base est sur l'axe des x, dont l'intérieur ne contienne aucun des n points et qui soit de surface maximale.

Par exemple si $h = 20, l = 25$, et qu'il y a 5 points $(2,5), (5,17), (11,4), (16,6), (20,1)$, voici quelques exemples de rectangles qu'on peut dessiner (voir figure 1) : $(5,0)(5,20)(11,20)(11,0)$ de surface 120, $(0,0),(0,1),(25,1),(25,0)$ de surface 25, $(0,0),(0,4),(20,4),(20,0)$ de surface 80, $(20,0),(20,20),(25,20)(25,0)$ de surface 100 ... La surface maximale qu'on peut obtenir est 153 avec le rectangle $(2,0)(2,17),(11,17)(11,0)$.

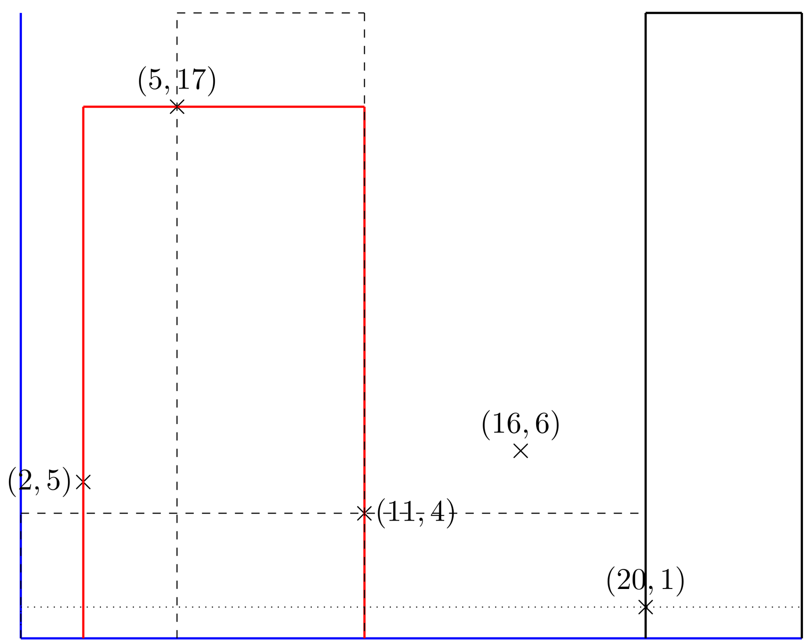

FIGURE 1 – Exemple de problème.

Donc les données en entrée du problème sont :
- l, h, deux entiers strictement positifs
- n, un entier positif
- les coordonnées entières des n points $(x_i, y_i), 0 \leq x_i \leq l, 0 < y_i < h$.

La sortie attendue est la surface maximale d'un rectangle vérifiant les contraintes.

Pour vérifier que vos codes sont corrects, de « petits » exemples avec solutions sont disponibles dans le répertoire

`Verif` (sur les machines de la PPTI) sous :
`/Infos/lmd/2016/master/ue/PSFPN_PFortin/JeuxDeDonnees`

Le répertoire `Perf` contient des jeux de données générés aléatoirement, ainsi que le code C effectuant cette génération. Vous pouvez utiliser ces jeux de données pour vos tests de performance, et si nécessaire générer des jeux de données supplémentaires. Pour vous aider dans la conception et l'écriture des algorithmes, dans tous ces exemples les points sont triés par x croissants, et le premier point correspond à $x = 0$, et le dernier à $x = l$.

2 Algorithmes naïfs

Une première approche peut être basée sur la remarque suivante : un rectangle de surface maximale respectant les contraintes a nécessairement deux sommets de la forme $(x_i, 0)$ et $(x_j, 0)$ avec $0 \leq i < j \leq n - 1$. Pourquoi ?

Ecrire une fonction qui prend en paramètre 2 points x_i et x_j et qui calcule le rectangle de surface maximale ayant son côté gauche en $x = x_i$ et son côté droit en $x = x_j$, et respectant les contraintes.

Déduisez en un algorithme en $O(n^3)$. Implémentez le et mesurez ces performances.

Comment peut-on obtenir un algorithme en $O(n^2)$? Implémentez le et mesurez ces performances.

Vous pourrez manipuler les coordonnées via des entiers non signés sur 32 bits mais les calculs et le résultat devront utiliser des entiers non signés sur 64 bits.