



Projet individuel: rédaction d'un rapport de recherche sur l'algorithme Toussaint résolvant le problème du rectangle minimum.

BM Bui-Xuan

1 L'énoncé du projet

Il s'agit d'analyser la qualité en tant que conteneur du calcul Toussaint d'un rectangle couvrant un ensemble de points dans le plan. L'évaluation se fera en fonction de la lisibilité du code, et surtout (à lire : 90%) de la rédaction du rapport.

Exercice 1

1. Implanter l'algorithme Toussaint vu en cours.
2. Récupérer une base de test de taille conséquente : le nombre minimum d'instances de test dans la base de test doit être 1664 ; le nombre minimum de points dans une instance doit être 256. Par exemple, une base de test vérifiant ces contraintes est celle de VAROUMAS disponible en ligne à l'adresse suivante :
http://www-apr.lip6.fr/~buiquan/files/algav2019/Varoumas_benchmark.zip
3. Expérimenter l'algorithme Toussaint avec les instances de test de la base, p.e. celle de VAROUMAS. Pour chaque instance on sera notamment intéressé par le rapport de qualité, en tant que conteneur, du rectangle minimum par rapport à l'enveloppe convexe selon la formule suivante :

$$\text{qualité} = \frac{\text{aire rectangle}}{\text{aire polygone}} - 100\%$$

Pour le calcul de l'aire du polygone on peut par exemple consulter la page

http://www.mathwords.com/a/area_convex_polygon.htm

4. Rédiger un rapport de cette expérience (introduction, résultats, discussion, conclusion) : on peut utiliser comme modèle le rapport de recherche suivant, qui traite une question similaire :
<http://hal.inria.fr/inria-00072354/PDF/RR-4233.pdf>

Exercice 2

1. Implanter l'algorithme Ritter vu en cours.
2. Confronter ce calcul avec la base de test, p.e. celle de VAROUMAS. Pour chaque instance on sera notamment intéressé par le rapport de qualité, en tant que conteneur, du cercle Ritter selon la formule suivante :

$$\text{qualité} = \frac{\text{aire cercle}}{\text{aire polygone}} - 100\%$$

3. Ajouter la nouvelle expérience au rapport.

Question bonus : Quelle est votre impression par rapport aux notions similaires en 3D ?

Remarques sur la rédaction du rapport

Un effort particulier doit être mis sur la phase de test et celle de la rédaction du rapport. On veillera à expliciter les points suivants, pour chaque algorithme implémenté, e.g. Toussaint et Ritter :

- définition du problème et la structure de données utilisée.
- analyse et présentation théorique des algorithmes connus dans la littérature.
- argumentation concise appuyant toute appréciation, amélioration, ou critique à propos de ces algorithmes existants dans la littératures.
- partie test : méthode d'obtention des *testbeds*. En particulier, il est important de citer la provenance des fichiers sources, e.g. celle de VAROUMAS si l'on avait décidé de l'utiliser.
- test de performance : mieux vaut privilégier les courbes, diagrammes bâton (moyenne + écart type) et diagrammes de fréquence, plutôt qu'exhiber les colonnes de chiffre sans fins...
- une discussion sur les résultats de test de performance est toujours la très bienvenue.
- conclusion et perspectives sur le problème de conteneur d'aire minimum.

Il est prudent d'avoir entre 8 et 15 pages pour un rapport avec un contenu moyen. Le nombre de pages recommandé pour ce rapport est 12 pages. Il convient de bien respecter cette limitation : les pages 15+ ne seront pas lues !

Contraintes :

- Le rapport est à écrire de façon individuelle. Le plagiarisme est strictement interdit.
- Archiver la totalité du rendu en un seul fichier compressé contenant de la documentation (rapport \approx 8-15 pages), un binaire (ou les `.class` si Java) et un README expliquant comment exécuter le binaire, le code source commenté, un Makefile (ou Apache Ant si Java), un répertoire contenant une certaine partie de la base de test (attention à la limitation en espace), et tout ce dont on juge utile à la lecture du projet sans toute fois dépasser la dizaine de Méga-octet.
- Envoyer ce fichier à `buixuan@lip6.fr`, 3 emails maximum par groupe. L'utilisation des hébergeurs en ligne (drive et compagnies) est proscrite. La nomination de préférence est soit `algav-projet-2019-NOM.piki`, où `piki` peut être un élément de $\{tgz, zip, rar, 7z, etc\}$.
- Deadline : 12 Janvier 2020, 23h59, cachet de serveur de messagerie faisant foi. Pénalité de retard : malus de $2^{h/24}$ points pour h heures de retard.