

INF591: Algorithmes du monde réel – TP feuille 1

L'objectif du TP est de programmer (en C ou C++) une heuristique pour le calcul d'une couverture par sommets (*Vertex Cover*) d'un graphe. Pour tester votre programme vous utiliserez le générateur de graphes `gengraph`, comme dans votre projet, dont le source peut être obtenu par le lien <http://dept-info.labri.fr/~gavoille/gengraph.c>. Le générateur peut être compilé avec `gcc gengraph.c -o gengraph`.

1 Structure de données

Tout d'abord il faut charger le graphe, donné sous la forme d'un fichier texte au format de `gengraph`¹, en mémoire, dans une structure de données adéquate pour ensuite appliquer votre heuristique.

Nous vous conseillons d'utiliser une liste d'adjacences. Par exemple, utiliser une liste de listes. Disons que $G[i]$ représente la liste d'adjacences pour le sommet i , numéro allant de 0 à $n - 1$, n étant nombre de sommets du graphe. Ainsi $G[i][k]$ donnera le k^e voisin du sommet i , k allant de 1 au degré de i (vous pourrez utiliser $G[i][0]$ pour stocker le degré de i).

2 Vérifier sa solution

Il est impératif de fournir une fonction, qui étant donné un graphe G (donné sous liste d'adjacences) et une liste C de sommets, indique si C est une couverture par sommets valide pour G .

3 Heuristique

L'heuristique que vous devez implanter est un algorithme glouton dit du "degré maximum". Dans la suite, la notation $G \setminus \{x\}$ correspond au graphe obtenu à partir de G en supprimant le sommet x et ses arêtes incidentes.

DEGMAXVC(G)

1. Si G n'a pas de sommet ou pas d'arête, alors retourner l'ensemble vide.
2. Choisir un sommet x de degré maximum dans G .
3. Calculer le graphe $H := G \setminus \{x\}$.
4. Retourner l'ensemble $\{x\} \cup \text{DEGMAXVC}(H)$.

¹Le format est décrit dans l'aide en ligne, en appelant `gengraph -help`.

4 Tests

Quelle est la taille de la couverture donnée par cette heuristique dans le cas d'un chemin à n sommets? Tester votre programme avec des chemins de taille croissante (en utilisant une commande du type `gengraph path ...`) et en déduire le comportement asymptotique (en fonction de n) de la taille de la couverture. Vous devriez obtenir une taille en αn . Que vaut α expérimentalement? Même question en utilisant l'option `-permute` de `gengraph` annulant l'effet de la numérotation des sommets lors du choix du sommet de degré maximum x .

5 Amélioration

Proposer un algorithme, et implanter celui-ci, donnant une couverture *optimale* pour un graphe composé d'une union disjointe de chemins, c'est-à-dire un chemin dans lequel manque certaine arête. Tester votre programme avec des graphes générés par `gengraph -dele 0.2 path ...` permettant de générer un chemin où chaque arête est supprimée avec probabilité $p = 0.2$. Pour cette probabilité $p = 0.2$, que vaut (en moyenne) la nouvelle valeur de $\bar{\alpha}$ lorsque n est grand? c'est-à-dire la constante de propotionalité de la taille d'une couverture par sommets optimale.

NB : Par défaut, les tirage aléatoires dans `gengraph` sont différents d'une exécution à l'autre. Pour « rejouer » le même graphe utilisez l'option `-seed x` où x est un entier que vous fixez.