

Algorithmes pour les graphes - TD sur plateforme

Exercice 5 : Implémentation d'une fonction *bound* plus sophistiquée

Une fonction d'évaluation plus évoluée (qui calcule une borne plus proche de la solution optimale, mais avec une complexité plus élevée) consiste à rechercher, pour chaque sommet de $nonVus$, la longueur de l'arête la plus courte permettant de le relier au circuit. Plus précisément :

- soit l la longueur du plus petit arc partant du dernier sommet visité (i.e., $vus[nbVus-1]$) et arrivant sur un des sommets non visités (i.e., de $nonVus[0..nbNonVus-1]$);
- $\forall i \in [0, nbNonVus-1]$, soit l_i la longueur du plus petit arc partant de $nonVus[i]$ et arrivant soit sur 0, soit sur un des sommets de $nonVus[0..nbNonVus-1]$;

une borne inférieure de la longueur du plus court chemin partant de $vus[nbVus-1]$, passant par chaque sommet de $nonVus[0..nbNonVus-1]$, et se terminant sur 0 est : $l + \sum_{i=0}^{nbNonVus-1} l_i$

Votre travail : Vous devez implémenter la procédure `permut` :

```
int permut(int vus[], int nbVus, int nonVus[], int nbNonVus, int longueur, int pcc)
```

telle que :

- le tableau `vus[0..nbVus-1]` contient les sommets de la liste `vus`,
- le tableau `nonVus[0..nbNonVus-1]` contient les sommets de l'ensemble `nonVus`,
- la variable `longueur` contient la longueur du chemin correspondant à `vus[0..nbVus-1]`,
- la variable `pcc` contient la longueur du plus court circuit trouvé depuis le début.

La postrelation de cette fonction est la même que pour l'exercice 3, mais vous ajouterez une étape d'évaluation pour réduire l'espace de recherche.

Code fourni (téléchargeable sur <http://liris.cnrs.fr/csolnon/TPTSP/code3.c>) : cf code fourni pour l'exercice 3.

Exemple d'exécution : Les temps CPU (en secondes) sont donnés à titre indicatif, pour un processeur 2,6 GHz Intel Core i5.

Entrée	Sortie	Temps CPU
4	72	0.00
6	91	0.00
8	123	0.00
10	134	0.00
12	161	0.00
14	182	0.00
16	198	0.00
18	230	0.02
20	261	0.35
22	281	0.26
24	299	1.09
26	313	12.48
28	326	10.16
30	349	46.25
32	361	61.24