

1. But de TP :

Le but est d'améliorer le processus de recherche du plus court chemin (problème de voyageur de commerce) en implémentant une fonction d'estimation.

2. Solution proposée :

Notre solution consiste à diviser les sommets voisins de dernier élément visité en deux « arrivée » et « départ ». Puis, pour chaque état on calcule la somme de $\text{min_colonne}/2$ (arrivée) avec le $\text{min_ligne}/2$ (départ).

```
// la fonction de sous-estimation h
int fonc_H( TypeChemin *e )
{
    if (e->ns==n){
        return poids[e->chem[e->ns-1]][0];
    }
    int h = ((double)minLigne(e->chem[e->ns-1])/2);
    int i;

    for(i=0;i<n;i++)
    {
        if (!Existe_dans_chemin(i,e))
        {
            h=h+(double) (minLigne(i)/2)+(double) (minColumn(i)/2);
        }
    }
    h=h+(double) (minColumn(e->chem[0])/2);

    return 0;
}

// fin de la fonction d'estimation h
```

3. Comparaison de notre solution avec PVC_dfs :

a. Résultat de PVC_BB avec sous-estimation :

```
1) Générer un graphe complet aléatoire
2) Lire le graphe à partir d'un fichier donné
Choix : 2
Nom du fichier contenant le graphe : fct.txt

matrice des poids des aretes:
      0  1  2  3  4  5  6  7
-----
0|  0  4  5  5  6  7  1  2
1|  4  0  5  1  5  5  4  5
2|  5  5  0  6  8  6  8  5
3|  5  1  6  0  1  1  2  6
4|  6  5  8  1  0  6  8  4
5|  7  5  6  1  6  0  3  5
6|  1  4  8  2  8  3  0  4
7|  2  5  5  6  4  5  4  0

PVC avec Branch and Bound pure (file de priorité de type Heap)

Début de Branch & Bound

Fin de Branch & Bound  Nb itérations = 2911  Taille max de la file = 3463
Le cycle hamiltonien de poids minimum est :
  0  7  4  3  1  2  5  6  0 Son cout est = 23

Process returned 0 (0x0)  execution time : 25.687 s
Press any key to continue.
```

b. Résultat de PVC_dfs :

```
1) Générer un graphe complet aléatoire
2) Lire le graphe à partir d'un fichier donné
Choix : 2
Nom du fichier contenant le graphe : fct.txt

matrice des poids des arêtes:
 0  4  5  5  6  7  1  2
 4  0  5  1  5  5  4  5
 5  5  0  6  8  6  8  5
 5  1  6  0  1  1  2  6
 6  5  8  1  0  6  8  4
 7  5  6  1  6  0  3  5
 1  4  8  2  8  3  0  4
 2  5  5  6  4  5  4  0

PVC avec dfs

Le cycle hamiltonien de poids minimum est :
 0  7  4  3  1  2  5  6  0 Son coût est = 23
Nb total de solutions = 5040
Temps d'exécution : 23.000000 ms

Process returned 0 (0x0)   execution time : 13.083 s
Press any key to continue.
```

- On remarque que le cycle hamiltonien de poids minimum et le coût de ce dernier sont les mêmes dans les deux méthodes (BB avec sous-estimation et dfs). Par contre, BB avec sous-estimation est plus efficace en termes de nombre d'itérations et la taille de la file que dfs.