Centre Universitaire de Mila

Institut des sciences et de la technologie

1ère Master STIC

Année : 2018-2019

Module :

Contrôle d’optimisation combinatoire et

métaheuristiques

**Exercice 1: 5.5 points**

1. Quelle est la différence entre une heuristique et une métaheuristique.
2. En utilisant comme exemple le problème de sac à dos, expliquer les caractéristiques des problèmes de la classe NP.
3. Soit P1 et P2 deux problèmes d’optimisation :

* Quand on peut dire que P1 est NP-complet ?
* Si P1 est NP-complet et P1 se réduit à P2. Qu’est-ce qu’on peut dire sur P2.

1. Nous cherchons à trouver le maximum d’une fonction *f* (x) sur l’intervalle [0, 128] en utilisant un algorithme génétique. Calculer la taille du chromosome (représentation binaire) dans le cas où *x* est un entier et dans le cas où *x* est un réel avec deux nombre après la virgule.

**h(n0)**

**2**

**3**

**0**

**2**

**5**

**2**

4

2

3

2

1

7

4

1

**9**

4

**Exercice 2: 3.5 points**

On cherche à trouver le plus court chemin entre le nœud n0 et n6.

1. Résoudre ce problème par l’algorithme A\*.
2. Est-ce que la solution trouvée est optimale ? justifier.

**Exercice 3: 5 points**

Dans un problème d’optimisation, nous avons un sac à dos de poids maximal P (p=9 kg) et n objets (n=5). Pour chaque objet i, nous avons un poids pi et un bénéfice vi. Le problème à résoudre consiste à choisir les objets à prendre parmi les n objets de manière à avoir un bénéfice maximal et respecter la contrainte du poids à ne pas dépasser. Chaque objet doit être sélectionné au plus une fois.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Objet | 1 | 2 | 3 | 4 | 5 |
| Vi | 20 | 30 | 35 | 12 | 3 |
| Pi | 2 | 5 | 7 | 3 | 1 |
| Vi/Pi | 10 | 6 | 5 | 4 | 3 |

1. Donner la formulation mathématique de ce problème.
2. En choisissant un parcours en profondeur, appliquer l’algorithme branch and bound pour la résolution de ce problème. Préciser la méthode de calcule de la borne supérieure pour les trois premiers nœuds dans l’arbre.

**Exercice 4: 6**

Soit le problème d’affectation suivant : nous avons n personnes à affecter à n tâches (n=4). Le coût de l’affectation de la personne i à la tâche j est noté par cij. Le problème consiste à affecter chaque personne à une seule tâche de telle manière à minimiser le coût total. Une tâche ne peut être faite que par une seule personne. La matrice des coûts est la suivante.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Taches  Personnes | T1 | T2 | T3 | T4 |
| P1 | 9 | 2 | 7 | 8 |
| P2 | 6 | 4 | 3 | 7 |
| P3 | 5 | 8 | 1 | 8 |
| P4 | 7 | 6 | 9 | 4 |

Nous considérons que la solution initiale ***s*** est (T1🡨P1, T2🡨P3, T3🡨P4, T4🡨P2) et que le voisin d’une solution est généré par la permutation de l’affectation entre une tache i et une tache (i+1) mod *n*.

1. Si on utilise la méthode de recuits simulés avec T=10. Calculer pour chacun des voisins de la solution s la probabilité d’être choisi comme solution courante.
2. En appliquant la méthode de recherche taboue, donner pour deux itérations différentes : L’état de la liste taboue, l’ensemble N(s) généré, la solution choisie et son évaluation et la meilleure solution.
3. Après la définition des types de données nécessaires, écrire en algorithme la fonction f(s, M) permettant d’évaluer une solution s où M est la matrice des coûts.