Le 25 Juin 2019, durée 2 heures UTBM

Final AG41 – Printemps 2019

Durée 2 heures – Documents autorisés - PC **non** autorisés

Exercice 1 : Algorithmes évolutionnaires (6 pts)

Soit le problème de voyageur de commerce suivant : Soit V={v1,v2,v3,v4,v5} 5 villes à parcourir. Un algorithme évolutionnaire a été développé pour rechercher le cycle Hamiltonien (visite de chaque ville une et une seule fois) de plus court chemin. Les solutions du problème sont codées par une permutation définissant l'ordre de passage sur les villes. Ex. La solution <2,3,1,5,4> veut dire que les villes ont été visitées dans l'ordre v2, v3, v1, v5 et puis v4. Les distances inter-ville sont données par le tableau suivant :

	v1	v2	v3	v4	v5
v1	0	12	3	23	1
v2	12	0	9	18	3
v3	3	9	0	89	56
v4	23	18	89	0	87
v5	1	3	56	87	0

1- Quelle est l'évaluation de la solution <1,5,2,4,3> et la solution <5,4,3,2,1> ?

2- A chaque génération, la probabilité de sélection d'un individu s de la population courante est égale à :

$$P(s) = \frac{F(s)}{\sum_{y \in Population} F(y)}$$

F(s) : Fonction de cout.

- Que pensez-vous de cette formulation ? Pose-t-elle un problème, lequel ?
- Si oui, proposez une formulation de P(s) pour remédier au problème.

3- On combine deux solutions en les coupant de manière aléatoire, au même endroit, et en recollant les morceaux. On appelle ce type de croisement « croisement mono-point ».

Parent 1	4 5 1 2 3	⇉	Enfant 1	4 5 1 3 5
Parent 2	4 2 1 3 5		Enfant 2	4 2 1 2 3

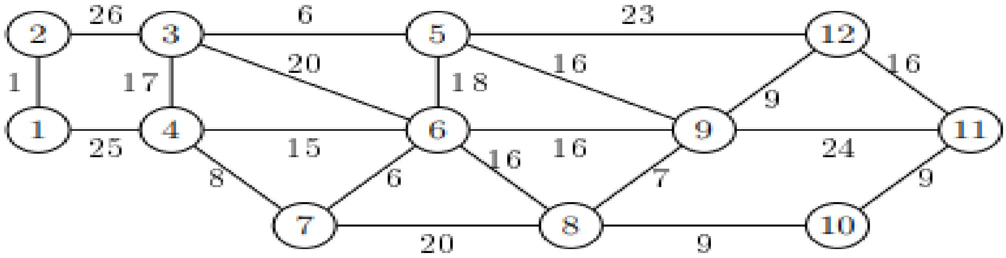
- Quel problème pose cet opérateur ?
- Proposé un opérateur de croisement remédiant à ce problème (écrire l'algorithme).

4- Proposer un opérateur de mutation pour le problème de voyageur de commerce (écrire l'algorithme).

5- On fixe à chaque génération, la probabilité de mutation d'un individu S de la population à 1, cela signifie quoi et que déduisez vous ?

Exercice 2 : Recherches tabou (14 pts)

Arbre-k minimum Il s'agit d'extraire d'un graphe un arbre de poids minimum(la somme des poids des arêtes est minimum) contraint à ne posséder que k arêtes, le problème est Np-difficile. On prendra pour exemple le graphe suivant avec k = 4.



1- Construire une solution initiale, k=4, point de départ 1.
Piste : On peut choisir, à chaque étape, un minimum local en sélectionnant l'arête frontière de poids minimum parmi les arêtes frontières.

2- Définir une opération de voisinage Il s'agit de définir l'opération qui, à partir d'une solution, détermine des solutions voisines.
Piste : On peut utiliser une opération d'échange d'arêtes.

3- Deux listes tabou formées des arêtes :
- Arêtes ajoutées : dire d'une arête ajoutée qu'elle est taboue, c'est dire qu'il est interdit de la retirer
- Arêtes retirées : dire d'une arête retirée qu'elle est taboue, c'est dire qu'il est interdit de l'ajouter

Existe-t-il une limite à la durée pendant laquelle une arête ajoutées peut rester taboue ?

4- On prend comme solution initial {(1, 2),(1, 4),(4, 7),(6, 7)}, Compléter le tableau suivant :

Itération	Liste taboue		ajout	retrait	poids
	1	2			
0	∅	∅	∅	∅	40
1	∅	∅	(4,6)	(4,7)	47
2	(4,6)	(4,7)	(6,8)	(6,7)	57
3	(6,8) , (4,7)				
4					
5					
6					
7					38

5- Quelle est la solution trouvé après 7 itération ?

6- C'est quoi un critère d'aspiration et quand on l'utilise?

7- On désire déterminer intelligemment une solution à partir de laquelle relancer une nouvelle recherche :

- À partir de quelles solutions ne veut-on pas relancer la recherche ?
- On relance la recherche taboue à partir d'une nouvelle solution initiale, remplir le tableau (attention de commencer avec des optimums locaux).

Itération	Liste taboue		ajout	retrait	poids
	1	2			
0	∅	∅	∅	∅	
1	∅	∅			
2					
3					

8- Quelle est la valeur de fonction objectif (poids) de l'optimum globale?

Bon courage