Examen AI52

Exercice 1

On suppose qu'un algorithme génétique utilise des chromosomes de la forme $x = a\ b\ c\ d\ e\ f\ g\ h$ avec une taille fixe de 8 gènes. Chaque gène peut prendre n'importe quel digit entre 0 et 9. Le fitness à maximiser de chaque individu est calculé selon la formule : $f(x) = (a+b) - (c+d)\ +\ (e+f) - (g+h)$

a) On suppose que la population initiale contient les 4 individus suivants :

$$x1 = 6\ 5\ 4\ 1\ 3\ 5\ 3\ 2$$

$$x2 = 8\ 7\ 1\ 2\ 6\ 6\ 0\ 1$$

$$x3 = 2\ 3\ 9\ 2\ 1\ 2\ 8\ 5$$

$$x4 = 4\ 1\ 8\ 5\ 2\ 0\ 9\ 4$$

1. Evaluer le fitness de chaque individu puis arranger les selon l'ordre décroissant de fitness.
2. Croiser les deux meilleurs individus en utilisant un simple point de croisement au milieu des individus.
3. Croiser les deuxièmes et troisièmes meilleurs individus en utilisant deux points de croisement (points après b et f).

b) On suppose maintenant que la nouvelle génération contient les 4 individus résultants des opérations de croisement dans les parties précédentes.

1. Evaluer le fitness de la nouvelle population (moyenne). Est-ce que le fitness s'est amélioré ?
2. Quel pourrait être le chromosome représentant la solution optimale. Quelle est sa fonction fitness ?
3. Que faut -t-il faire pour que l'algorithme puisse trouver la solution optimale ?

Exercice 2

L'énoncé de ce problème fameux est simple: « Étant donné plusieurs objets possédant chacun un poids et une valeur et étant donné un poids maximum pour le sac, quels objets faut-il mettre dans le sac de manière à maximiser la valeur totale sans dépasser le poids maximal autorisé pour le sac? »

Les objets sont numérotés par l'indice i variant de 1 à n. Les nombres p_i et v_i représentent respectivement le poids et la valeur de l'objet numéro i. La capacité du sac sera notée W.

Exemple : Pour quatre objets ($n = 4$) et un sac à dos d'un poids maximal de 30 kg ($W = 30$), nous avons par exemple les données suivantes

Objets	1	2	3	4
Valeur	7	4	3	3
poids	13	12	8	10

Pour décrire la façon pour remplir le sac à dos, il faut indiquer pour chaque élément s'il est pris ou non. On peut utiliser un codage binaire : l'état du i-ème élément vaudra 1 si l'élément est mis dans le sac, ou 0 s'il est laissé de côté.

Dans notre exemple, une solution réalisable est de mettre tous les objets dans le sac à dos sauf le premier, nous avons donc $x_1 = 0; x_2 = 1; x_3 = 1; x_4 = 1$

La contrainte du problème est évidente : la somme des poids de tous les objets dans le sac doit être inférieure ou égale au poids maximal du sac à dos. La fonction à maximiser ici est la valeur totale des objets dans le sac

1. **Cas n quelconque**. Donner l'expression mathématique de la fonction à optimiser ainsi que l'expression mathématique de la contrainte du problème

   Une façon de coder une solution (remplir le sac donc) est complètement décrite par un vecteur appelé vecteur contenu : $X = (x_1, x_2, x_3, \dots x_n)$

2. **Cas n= 4**

a. Existe-t-il des solutions non réalisables, si oui lesquelles ?

b. On souhaite appliquer la méthode Tabou en partant du sommet 0010. Donner les solutions voisines ainsi que leurs fitness.

c. Quelle taille de liste tabou doit-t-on prendre pour éviter un blocage ? Justifiez votre réponse.

Notation : t_{01}^{i} (respectivement t_{10}^{i}) la transformation consistant à changer le ième bit de 0 en 1 (resp de 1 en 0)

Exercice 3

Soit l'application g de $\mathbb{R}^2$ dans $\mathbb{R}$ définie par :

$$g(a,b) = \frac{a^4+b^4}{4} - \frac{3a^2b^2}{2} + \frac{a^2-b^2}{2}$$

1. Donner la matrice Hessienne $H(a,b)$
2. Quelles sont les points critiques de g et donner leurs natures.
3. Rappeler l'algorithme de Newton et exécuter une première itération en partant du point initial $x_0 = \left(0, \frac{\sqrt{3}}{3}\right)$ puis en partant du point $x'_0 = \left(0, -\frac{\sqrt{3}}{3}\right)$