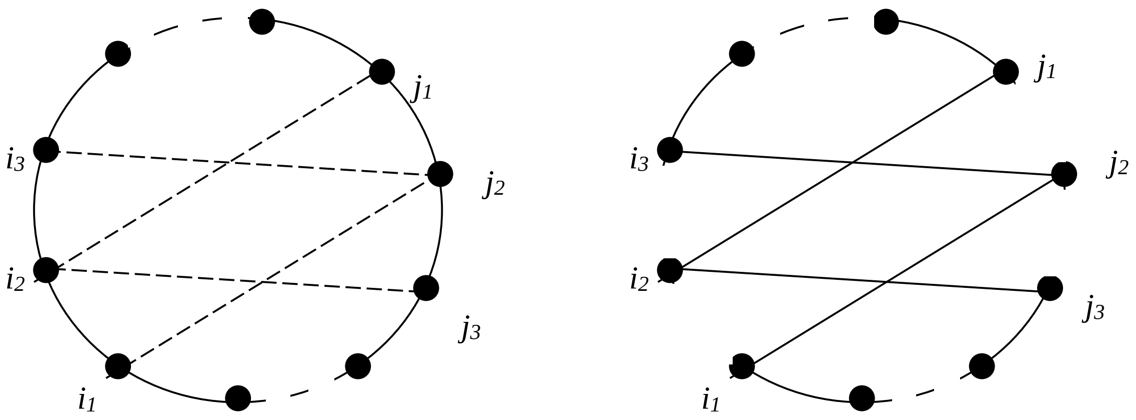
NOM SIGNATURE

Les documents de cours, TD et TP sont autorisés

Problème 1 – Voisinage : voyageur de commerce orienté (7)

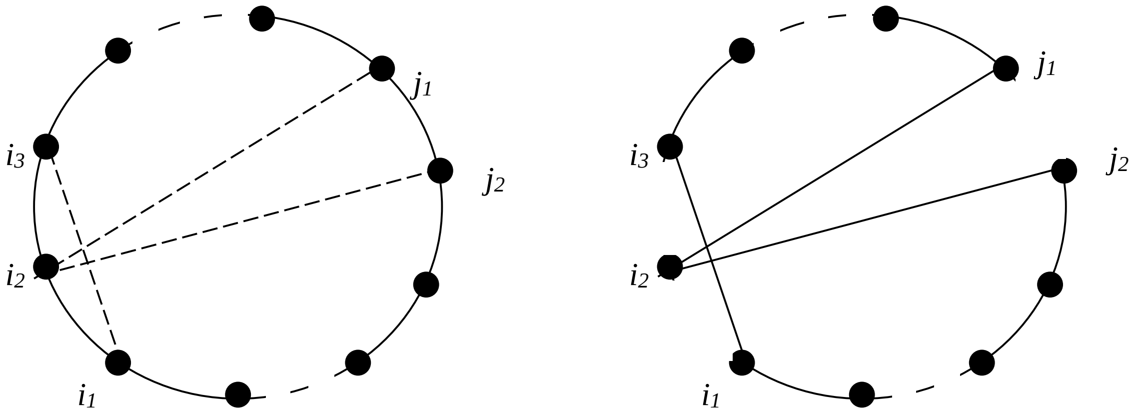
Considérons un graphe orienté $G = (X, U)$ avec un coût c_{ij} pour l'arc (i,j) de U. A partir d'un cycle Hamiltonien (cycle orienté qui passe une et une seule fois par chaque sommet de G), on définit deux transformations d'un cycle T1 et T2 représentées ci-dessous.

1. Déterminer la taille des voisinages de T1 sachant que les sommets i_1, i_2 et i_3 se suivent, que les sommets j_1, j_2 et j_3 se suivent, et que les sommets i et j sont tous différents. On notera $n=|X|$ le nombre de sommets. <u>Expliquer le raisonnement</u>.



Transformation T1 : Les arcs (i_1,i_2), (i_2,i_3), (j_1,j_2) et (j_2,j_3) sont remplacés par les arcs (i_1,j_2), (j_2,i_3), (j_1,i_2) et (i_2,j_3).

2. Déterminer la taille des voisinages de T2 sachant que les sommets i_1, i_2 et i_3 se suivent, que les sommets j_1, j_2 et j_3 se suivent, et que les sommets i et j sont tous différents. On notera $n=|X|$ le nombre de sommets. Expliquer le raisonnement.

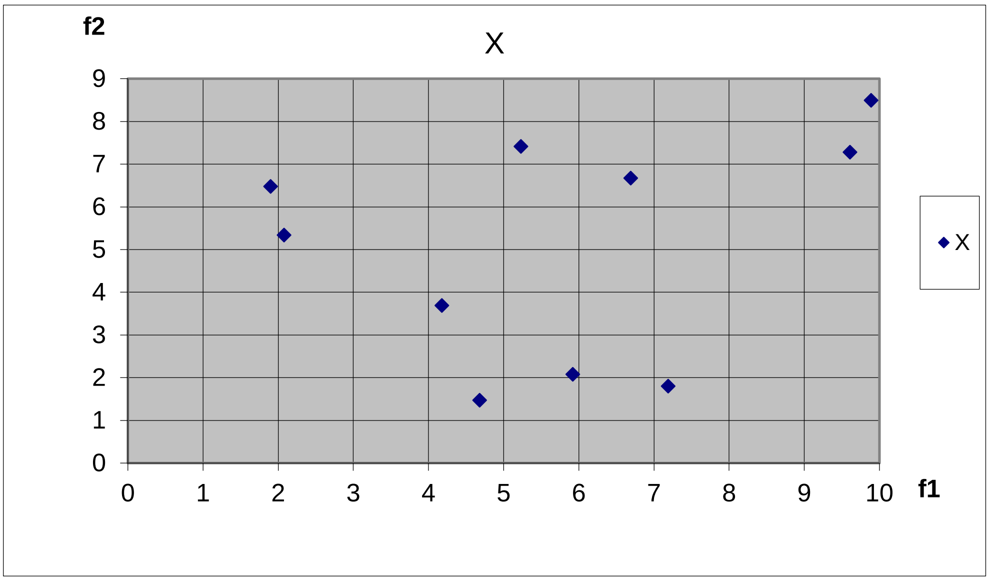


Transformation T2 : Les arcs (i_1,i_2), (i_2,i_3) et (j_1,j_2) sont remplacés par les arcs (i_1,i_3), (j_1,i_2) et (i_2,j_2).

Problème 2 – Multicritère : algorithme génétique bi-critère (8)

Un algorithme génétique bi-critère, où *f1* et *f2* sont les fonctions de coût à minimiser, obtient la population *X* de solutions avec les coûts suivants :

	x1	*x2*	*x3*	*x4*	*x5*	*x6*	*x7*	*x8*	*x9*	*x10*
f1	2.08	5.92	4.18	6.69	7.19	9.89	5.23	9.61	1.90	4.68
f2	5.34	2.08	3.69	6.67	1.80	8.49	7.41	7.28	6.48	1.47



1. Donnez l'ensemble des solutions qui dominent (5.0, 5.0).

2. Donnez l'ensemble des solutions dominées par (5.0, 5.0).

3. Donnez l'ensemble des solutions Pareto (non-dominées) de cette population.

4. Les solutions de X sont classées avant d'être utilisées par l'itération suivante de l'algorithme. Calculez le rang des solutions de la population selon la méthode *NSGA* et *NDS*.

Sélection NSGA (Nondominated Sorting Genetic Algorithm) : cette méthode se base sur la détermination progressive des différents niveaux de dominance entre les solutions. Tout d'abord un ensemble *I* est initialisé à l'ensemble de toutes les solutions de la population. Les individus non-dominés dans *I* obtiennent le rang *i*=1 et sont retirés de l'ensemble *I*. Ce cycle est répété jusqu'à ce que l'ensemble *I* soit vide en incrémentant le rang à chaque itération *i*=*i*+1.

Sélection NDS (Non Dominated Sorting) : le rang d'une solution est égal au nombre de solutions qui la dominent dans la population, plus 1.

	x1	*x2*	*x3*	*x4*	*x5*	*x6*	*x7*	*x8*	*x9*	*x10*
Rang NSGA										
Rang NDS										

5. On considère que chaque solution x_i de la population a une probabilité d'être sélectionné :

$$P_i = \frac{(TP - r_i)}{\sum_{j \in pop} (TP - r_j)}$$

Calculez la probabilité de sélection de chaque solution de la population selon les méthodes *NSGA* et *NDS*.

	x1	*x2*	*x3*	*x4*	*x5*	*x6*	*x7*	*x8*	*x9*	*x10*
P pour NSGA										
P pour NDS										

6. Quelle conclusion tirez-vous de NSGA et NDS sur la convergence de l'algorithme génétique en comparant les distributions de probabilité ?

Problème 3 – Combinatoire : placement de reines (5)

Soit un problème de placement de reines de taille 10 (échiquier 10x10). On place 10 reines sur l'échiquier et une case ne peut contenir qu'une seule reine.
Les résultats doivent être démontrés.

1. Quel est le nombre de combinaisons possibles de placement ?

2. Quel est le nombre de combinaisons possibles de placement si la règle de placement des reines est de placer une seule reine par ligne ?

3. Quel est le nombre de combinaisons possibles de placement si la règle de placement des reines est de placer une seule reine par ligne et une seule reine par colonne ?

4. Définition : un conflit est un placement de 2 reines tel qu'elles se prennent mutuellement en ligne, en colonne ou en diagonale (exemple, si R1 et R2 se prennent mutuellement, on compte 1 seul conflit).

   Ecrivez mathématiquement une fonction de coût qui compte le nombre de conflits entre les reines placées sur un échiquier de taille N quelconque. On utilise la notation suivante : le problème est modélisé par une matrice $X[i,j]$ où $x_{ij}=1$ si une reine est placée sur la case de ligne i et de colonne j sinon $x_{ij}=0$.