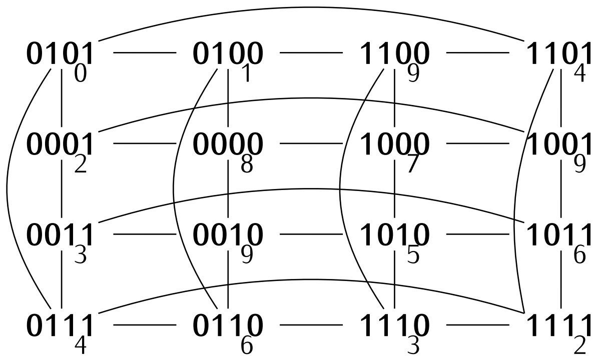
TDs : Méta-heuristiques

Exercice 1 (Test des algorithmes de recherche). On considère un exemple jouet dont l'espace de recherche est donné à la figure ci-contre. Chaque solution est représentée par un nombre en binaire. Le voisinage d'une solution est l'ensemble de toutes les solutions qui diffèrent de celle-ci d'un seul bit. En dessous de chaque solution est indiquée la valeur de la fonction dont on cherche un minimum.



Question 1.1. On rappelle ci-contre l'**Algorithme générique de recherche locale** : Faire un "essai" de recherche locale en partant de la solution 1000 en appliquant la méthode de **descente selon la plus grande pente** (inverse du Steepest Hill Climbing) :

> • on part d'une **solution** s
> Repeat
> | • choisir un **voisin** s' de s
> | • si acceptable_voisin(s',s) alors $s \leftarrow s'$
> Until condition d'arrêt

- `choisir_un_voisin` renvoie un voisin s' t.q. $f(s') =$ min parmi tous les voisins de s ;
- `acceptable_voisin(s,s')` renvoie vrai si $f(s') < f(s)$;
- `condition_d'arrêt` = quand pas de voisin s'' t.q. $f(s') < f(s)$ (pas d'amélioration possible)

Que peut-on dire de la solution obtenue ?

Question 1.2. Faîtes un graphe dont les sommets sont les solutions, vous mettrez un arc (s, s') si s' est un meilleur voisin de s. Quelles sont les solutions à partir desquelles la méthode de descente selon la plus grande pente n'amènent pas à un minimum global ? Comment pallier ce problème ?

Question 1.3. On rappelle la méthode *Tabou* :

> • on part d'une solution s
> • Tabou $\leftarrow$ [] ; $msol \leftarrow s$; $nb\_depl \leftarrow 0$; $STOP \leftarrow false$
> Repeat
> | • if voisins_non_Tabou(s) $\neq \emptyset$ then $s' \leftarrow$ meilleur_voisin_non_Tabou(s)
> | else $STOP \leftarrow true$ /* plus de voisin non tabou*/
> | • Tabou $\leftarrow$ Tabou + $\{s\}$ /*tabou implémentée en FIFO de taille k*/
> | • if meilleur(s',$msol$) then $msol \leftarrow s'$ /* stockage meilleure solution courante*/
> | • $s \leftarrow s'$; nb_depl++
> Until nb_depl = MAX_depl or $STOP$
> Return($msol$)

Refaire un essai de recherche locale à partir de 1000 en utilisant cette fois une liste Tabou de taille 1 et un arrêt au bout de 8 itérations. La recherche s'arrête-t'elle sur le minimum ?

Question 1.4. *(Travail Personnel)* Même question en ne stockant dans la liste Tabou que les sommets meilleurs que tous leurs voisins non tabou, avec liste Tabou de taille 1 puis 3 et 10 essais.

Question 1.5. Il est parfois trop coûteux de stocker les solutions. Dans ce cas, on mémorise les mouvements qui ont mené à ces solutions. Notons qu'il faut mémoriser les mouvements inverses. Parcourez les solutions avec une liste tabou de taille 2 depuis 1011 en stockant dans cette liste tabou les mouvements qui reviennent vers un minimum local. On va donc enfiler dans la liste tabou les mouvements qui s'éloignent d'une solution minimale (c'est-à-dire ceux qui vont vers des solutions n'améliorant pas la solution courante (ce sont des mouvements qui permettent de sortir du minimum local donc il ne faut pas les défaire)). On peut noter les opérations 1, 2, 3, 4 (flip du ième bit (de gauche à droite)). Ici on se sert du fait qu'un mouvement est égal à son inverse.

Recuit Simulé *(Travail Personnel)*

Question 1.6. On considère la configuration $s = 1010$ et sa voisine $s' = 0010$, dans la méthode du recuit simulé, quelle est la probabilité que s' soit acceptée comme voisine de s en fonction de la température T ? Trouvez T pour que cette probabilité soit 1/2.

Question 1.7. Qu'en est-il de $s'' = 1110$?

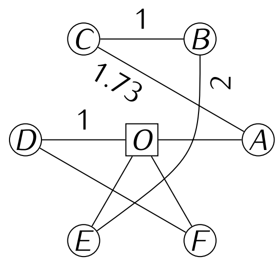
Question 1.8. Quelle température initiale de recuit simulé faut-il prendre pour avoir une probabilité ≥ 0.8 qu'une configuration quelconque soit acceptable depuis n'importe quelle solution initiale ? D'ordinaire il faut raisonner sur un échantillon de solutions prises au hasard mais ici on peut utiliser notre connaissance complète de l'espace des solutions.

Algorithme : Recuit simulé
1. $s \leftarrow s_0$ solution initiale
2. $T \leftarrow T_{MAX}$ température de départ
3. tant que $T > T_{\min}$ faire :
   (a) tant que équilibre pas atteint faire :
      i. $s' \leftarrow$ un voisin de s
      ii. $\Delta_E = f(s') - f(s)$. # *var. énergie*
      iii. si $\Delta_E < 0$: $p \leftarrow 1$; sinon $p \leftarrow e^{\frac{-\Delta_E}{T}}$;
      iv. acceptable $\leftarrow$ $\left|\begin{array}{l}\text{vrai avec prob. } p \\ \text{faux avec prob. } 1 - p\end{array}\right.$
      v. si acceptable : $s \leftarrow s'$
   (b) $T \leftarrow \lambda T$ avec $\lambda < 1$;

Exercice 2 (Routage de véhicules)**.** Une société de transport doit livrer des marchandises dans n villes $v_1, \ldots v_n$. Elle dispose de m camions $c_1, \ldots, c_m$ qui partent tous du même dépôt situé dans la ville O. On suppose qu'on connaît les distances entre les villes. Le but est de minimiser la distance totale parcourue par l'ensemble des m camions.

On considère une instance à 6 villes à visiter A à F avec 2 camions. Les villes sont disposées en hexagone régulier autour du dépôt O. Les distances sont données par la matrice ci-dessous. Le dessin indique une solution initiale définie par les trajets des deux camions : $S_0 = \{(ACBE), (DF)\}$.

	O	A	B	C	D	E	F
O		1	1	1	1	1	1
A			1	1.73	2	1.73	1
B				1	1.73	2	1.73
C					1	1.73	2
D						1	1.73
E							1



Question 2.1. Donnez une solution optimale.

Question 2.2. Les camions sont interchangeables donc pour représenter une solution on choisira de classer les camions par nombre de villes desservies décroissant. Quel est l'espace des solutions pour ce problème et quelle est sa taille ?

Question 2.3. Proposez une définition de voisinage utilisable pour ce problème dans le cas général de m camions et n villes. Il est intéressant que le voisinage soit simple à calculer et permette de couvrir l'espace des solutions. Pour cela répondez aux trois questions suivantes :
1. Quelle est la complexité du calcul de tous les voisins d'une solution ?
2. Donnez un exemple de deux solutions du problème avec 2 camions et 6 villes entre lesquelles il faut faire au moins 4 déplacements (si votre voisinage le permet).
3. Y-a-t'il des solutions inaccessibles par votre voisinage ?

Question 2.4. Combien faut-il de mouvements pour arriver à un optimum (local ou global) à partir de S_0 avec votre notion de voisinage sur cet exemple (méthode hill-climbing glouton) ?

Question 2.5. Que peut-on stocker dans une liste tabou sur ce type de problème pour s'autoriser à sortir des optimums locaux ?

Question 2.6. *(Travail Personnel)* En pratique, la solution optimale trouvée ci-dessus n'est souvent pas réalisable car les camions ont une capacité limitée (pour simplifier on traduit cette capacité en terme d'un nombre maximum k de villes désservables par camion). Écrivez un algorithme qui permet de générer une solution initiale aléatoire mais respectant cette contrainte de capacité.