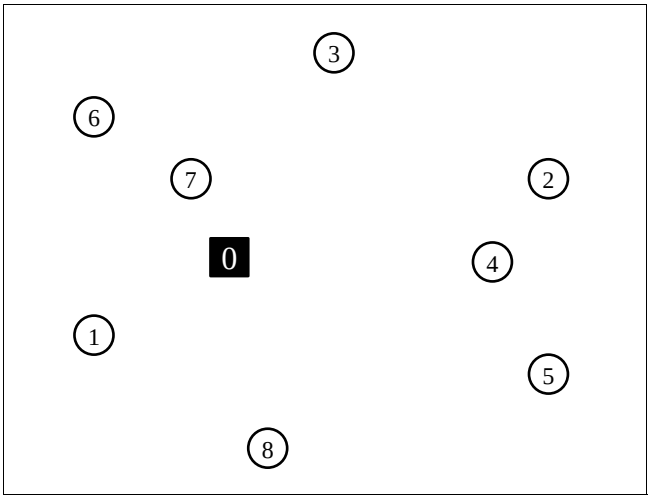
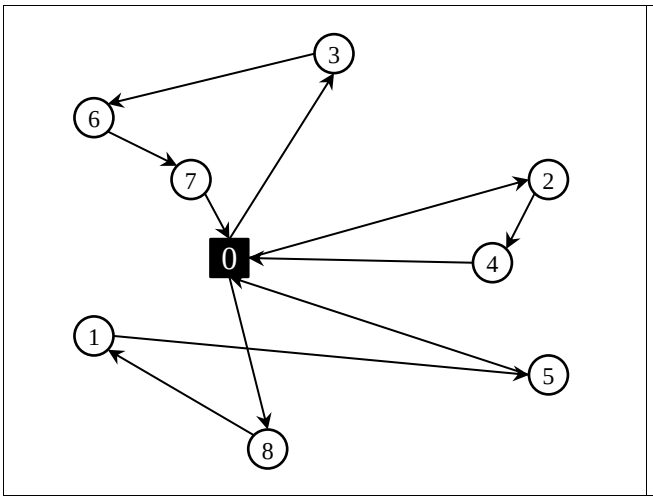
Recherche Opérationnelle
Duhamel - Lacomme
ISIMA 2F2 - 2F3

TP 3 : VRPTW
Durée : 3 séances, noté

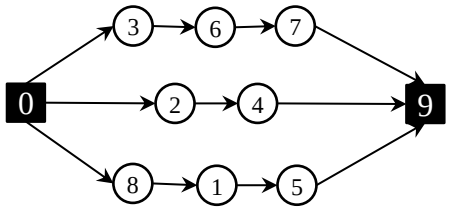
Présentation du problème : le problème de tournées de véhicules avec fenêtres de temps (*Vehicle Routing Problem with Time Windows* – VRPTW) consiste à définir un ensemble de routes de manière à satisfaire la demande d'un ensemble de n clients. Une flotte de véhicules est disponible en un dépôt central noté 0 pour effectuer ces livraisons. La flotte est homogène, *i.e.* chaque véhicule a la même capacité Q. La demande d_i de chaque client $i = 1 \dots n$ doit être traitée en une seule fois et par un seul véhicule. De plus chaque client $i = 1 \dots n$ possède une fenêtre de temps $[A_i; B_i]$ (heure d'ouverture A_i et de fermeture B_i) dans laquelle il doit être livré. Si le véhicule arrive avant l'heure d'ouverture, il doit attendre que le client ouvre. Par contre, il ne peut arriver après l'heure de fermeture. En outre, la livraison dure un temps t_i. La fenêtre de temps du dépôt $[A_0; B_0]$ correspond à la date au plus tôt de sortie du dépôt et la date au plus tard de retour au dépôt. La distance c_{ij} pour aller du client i au client j est connue. On fait souvent l'hypothèse qu'elle est symétrique et satisfait l'inégalité triangulaire. L'objectif principal est de minimiser le nombre de véhicules puis, en critère secondaire, la distance totale parcourue. Ce problème est une des extensions principales du VRP et il est NP-difficile.



représentation géométrique d'une instance



représentation géométrique d'une solution



représentation logique d'une solution

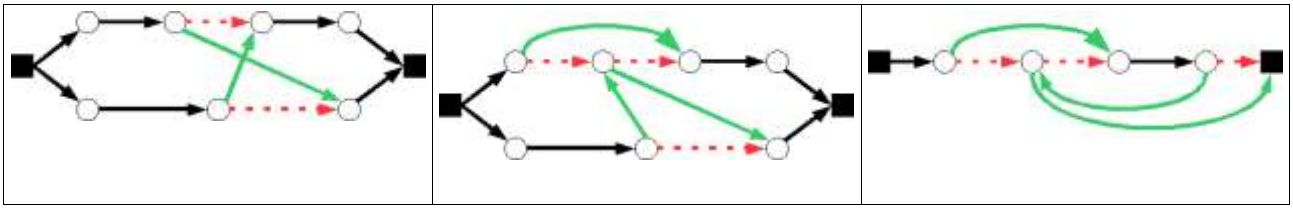
Il existe plusieurs manières de représenter une instance. Ici, la première permet de visualiser l'implantation des clients, de visualiser les tournées physiques et de voir rapidement si la solution est cohérente. La représentation logique présente les tournées de manière linéaire, en dupliquant le dépôt en une copie de sortie (noté 0) et une copie de retour (notée $n + 1$) des véhicules.

1) Récupération des instances
   Un jeu d'essai utilisé couramment sur le VRPTW a été proposé par M. Solomon en 1987. On peut le retrouver à la page de l'auteur (http://w.cba.neu.edu/~msolomon/problems.htm) ou à (http://www.sintef.no/Projectweb/TOP/VRPTW/Solomon-benchmark/100-customers/) avec les meilleures solutions trouvées. Chaque point (dépôt, livraison, collecte) possède des coordonnées X/Y et la distance entre deux points est la distance euclidienne.

2) Manipulations élémentaires
   Définir les structures de données adaptées au stockage des données et à la manipulation efficaces des solutions. La représentation de la solution nécessite de stocker un ensemble de tournées et, pour chacune, la date de passage du véhicule sur chaque point (pour accélérer les calculs de réalisabilité temporelle). On réfléchira aux informations pertinentes à stocker sur chaque sommet de manière à faciliter le plus possible les transformations sur la solution courante.

3) Heuristiques de construction
   Les contraintes temporelles compliquent la génération de solution réalisable. Par contre, le nombre de véhicule est libre. Voici deux heuristiques possibles :
   a) réaliser une routine construisant une solution à partir d'une séquence des clients. Une approche simple consiste à « ouvrir » une tournée et à insérer au mieux possible le client courant dans cette tournée. Lorsqu'il n'est plus possible d'insérer le client dans cette tournée, on ouvre une nouvelle tournée et on continue. Une amélioration considère l'insertion du client courant dans toutes les tournées ayant été ouvertes jusqu'à présent.
   b) l'heuristique *Savings* de Clarke & Wright a été développée pour le VRP. Elle ne change pas beaucoup pour le VRPTW, mis à part le test de réalisabilité temporelle. On crée initialement une tournée pour chaque client. Pour tout arc (i, j), on calcule le gain $g_{ij} = c_{i0} + c_{0j} - c_{ij}$. On trie les gains par ordre décroissant. Ensuite, dans cet ordre, g_{ij} étant le gain courant, on regarde s'il existe une route $(0, j, \dots, 0)$ commençant par le client j et une route $(0, \dots, i, 0)$ terminant par le client i. Dans ce cas, on fusionne les deux routes en supprimant les arcs $(0, j)$ et $(i, 0)$ et en ajoutant l'arc (i, j).

4) Heuristiques d'amélioration
   Le principe d'amélioration est identique à celui vu en cours. Les tests de validité sont plus complexes car ils doivent aussi prendre en compte la capacité et la compatibilité temporelle. Quelques exemples de mouvements intéressants :
   a) le 2-Opt* (ou cross) consiste à échanger les derniers clients de deux tournées
   b) le Or-Opt consiste à déplacer une séquence de clients (1, 2 ou 3) dans une autre tournée
   c) au sein d'une tournée, les opérations de déplacement d'un ou plusieurs sommets (*Shift*, *Swap*) permettent d'optimiser localement la tournée.



2-Opt* Or-Opt Shift

Les mouvements ci-dessus sont valides pour le TSP. Il faut les adapter pour le VRPTW.

5) Métaheuristiques
   Choisir une des métaheuristiques vues en cours. Multistart / GRASP et algorithmes génétiques / mémétiques sont de bons candidats. Attention à l'efficacité de la recherche locale. Une bonne recherche globale aura du mal à compenser une mauvaise recherche locale.
6) Analyse comparative
   Dans la littérature, il y a un flou dans la définition de la fonction objectif. Certains auteurs considèrent le nombre de véhicules comme un second critère. Etudier l'impact des fonctions objectif suivantes :
   - Le nombre de véhicules (critère principal) et la distance totale (critère secondaire)
   - La distance totale
   - La pondération du nombre de véhicules (coefficient α) et de la distance totale (coefficient $1 - \alpha$)
   - Un coût fixe K pour chaque véhicule plus la distance totale.

Modalités de remise du travail : comme pour le travail précédent, nous attendons un rendu contenant les sources du programme, le rapport décrivant les choix réalisés et présentant les tableaux de résultats. Ces tableaux sont fournis en annexe et sont à remplir avec vos résultats. Le programme principal doit accepter en entrée le nom de l'instance à traiter. Il doit ressortir une ligne contenant le temps de calcul et la valeur de la meilleure solution trouvée (nombre de véhicules et distance totale).

Bibliographie : quelques articles de référence

- Recherche Tabou : Potvin, J.-Y., T. Kervahut, B.L. Garcia and J.-M. Rousseau (1996), "The Vehicle Routing Problem with Time Windows Part I: Tabu Search", *INFORMS Journal on Computing* 8, 157–164.
- Recuit simulé : Chiang, W.C. and R.A. Russell (1996), "Simulated Annealing Metaheuristics for the Vehicle Routing Problem with Time Windows", *Annals of Operations Research* 63, 3-27.
- Algorithme génétique : Potvin, J.-Y. and S. Bengio (1996), "The Vehicle Routing Problem with Time Windows Part II: Genetic Search", *INFORMS Journal on Computing* 8, 165-172
- VNS : O. Bräysy (2003), "A Reactive Variable Neighborhood Search for the Vehicle-Routing Problem with Time Windows", *INFORMS Journal on Computing* 15(4): 347-368.
- GRASP : Kontoravdis, G.A. and J.F. Bard (1995), "A GRASP for the Vehicle Routing Problem with Time Windows", *Journal on Computing* 7, 10-23.
- Ant colony : Gambardella, L.M., E. Taillard and G. Agazzi (1999), "MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows", in *New Ideas in Optimization*, D. Corne, M. Dorigo and F. Glover (eds), 63–76, McGraw-Hill, London.
- LNS : Ropke, S. and D. Pisinger (2006), "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows", *Transportation Science* 40(4): 455-472.

Tableaux de résultats : il y a trois types d'instances, R, C et RC. Le premier type correspond à des clients répartis aléatoirement sur le plan. Le second correspond à des clients répartis en clusters (par exemple des villes). Le troisième est une combinaison des deux autres (clients en ville ou en campagne). Les instances contiennent 100 clients. Pour obtenir les instances à 25 et 50 clients, il suffit de considérer les 25 premiers ou les 50 premiers. Vous ferez des tableaux de résultats sur le modèle des tableaux pour le travail précédent.