

Recherche Opérationnelle - APP 1

Transports et réseaux

Sophie Demasse

26 septembre 2011

Références

- [1] David L. Applegate, Robert E. Bixby, Vasek Chvátal, and William J. Cook. *The Traveling Salesman Problem : A Computational Study*. Princeton University Press, 2006.
- [2] G. B. Dantzig, R. Fulkerson, and S. M. Johnson. Solution of a large-scale traveling salesman problem. *Operations Research*, 2 :393–410, 1954.
- [3] S. Lin and B. Kernighan. An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 21 :498–516, 1973.
- [4] I Or. *Traveling salesman-type combinatorial problems and their relation to the logistics of regional blood banking*. PhD thesis, Northwestern University, Evanston, Ill., 1976.
- [5] M.W.P. Savelsbergh. Local search in routing problems with time windows. *Annals of Operations Research*, 4 :285–305, 1985.
- [6] <http://www.tsp.gatech.edu/methods/>.

1 Première partie : Flots et couvertures

Convention de notations

Par la suite, on s'efforcera d'employer les notations suivantes :

- un graphe orienté simple est noté $G = (V, E)$ avec V l'ensemble des sommets, $|V| = m$ et $E \subseteq V \times V$ l'ensemble des arcs, $|E| = n$;
- $\delta^-(i)$ (respectivement $\delta^+(i)$) dénote l'ensemble des prédécesseurs (resp. successeurs) de $i \in V$;
- si le graphe G est valué, le poids (ou le *coût* ou la longueur) d'un arc $(i, j) \in E$ est noté $p_{ij} \in \mathbb{R}$; G est alors noté (V, E, p) ;
- si G est un *réseau de transport*, la *capacité* d'un arc $(i, j) \in E$ est notée c_{ij} , avec $c_{ij} \in \mathbb{Z}^+$; soient $s \in V$ (*source*) et $t \in V \setminus \{s\}$ (*puits*), deux sommets particuliers de G , on note G^0 le graphe obtenu de G en supprimant tout arc entrant en s et sortant de t et en ajoutant un arc (t, s) (*l'arc de retour*) ; la valeur d'un flot sur un arc $(i, j) \in E$ (le *flux* dans l'arc) est notée $f_{ij} \in \mathbb{R}_+$; $l_{ij} \in \mathbb{Z}_+$ dénote la borne inférieure du flot sur l'arc (i, j) et $a_{ij} \in \mathbb{R}_+$ dénote le coût d'une unité de flot sur l'arc.

Glossaire

1. Soit $G = (V, E, l, u)$ un graphe orienté, avec bornes inférieures et capacités sur les arcs ($0 \leq l_e \leq u_e, \forall e \in E$). À l'aide des notations précédentes, définir formellement les notions de graphes suivantes :
 - (a) un **flot compatible** dans G :
 - (b) un **flot [compatible] de s à t** dans G :
 - (c) la **valeur d'un flot** de s (*source*) vers t (*puits*) dans G :
 - (d) une **coupe** séparant s et t dans G :
 - (e) la **capacité** d'une coupe dans un graphe valué $G = (V, E, p)$:
 - (f) le **graphe d'écart** d'un flot :
 - (g) un **chemin augmentant** d'un flot :
2. Donner une définition des problèmes classiques d'optimisation suivants en précisant l'instance en entrée et la solution en sortie :
 - (a) problème de flot maximum
 - (b) problème de flot de coût minimum
 - (c) problème de coupe de capacité minimum
 - (d) problème de transbordement
 - (e) problème de couplage de cardinalité maximum
 - (f) problème de couplage de poids maximum
3. Définir formellement les notions suivantes :
 - (a) matrice **unimodulaire** :
 - (b) matrice **totalelement unimodulaire** :

1.1 Séance 1

Question 1 compléter ¹ le glossaire ci-avant.

Problème 1

La compagnie Nitrobec fabrique, à partir de nitrocellulose, un explosif industriel utilisé dans les travaux de génie civil. Le procédé de Nitrobec requiert des solvants très inflammables, et les explosifs obtenus doivent se manipuler avec grand soin. Le souci de réduire les risques a amené Nitrobec à répartir sa production entre 2 petites usines, T et U, installées dans la région portneuvoise à l'écart des habitations et accessibles par des routes dédiées. Nitrobec s'approvisionne en nitrocellulose auprès de 2 fournisseurs, F et G, qui lui consentent des prix que justifie l'importance de ses commandes. Nitrobec vient d'acheter 450 tonnes de nitrocellulose du premier fournisseur et 495 tonnes du second.

Le tableau 7.1 donne les coûts de transport pour acheminer la nitrocellulose des fournisseurs aux usines. Ces coûts sont assumés par Nitrobec. La capacité de production de chacune des usines est amplement suffisante pour traiter toute la nitrocellulose achetée.

TABLEAU 7.1 Coûts de transport de la nitrocellulose (en \$/t)	Fournisseur	Usine T	Usine U
	F	9	10
	G	8	11

Nitrobec considère qu'une tonne de nitrocellulose donne une tonne d'explosifs : la perte de poids subie par la nitrocellulose lors des traitements en usine est à peu près compensée par le poids des produits chimiques ajoutés pour l'obtention des explosifs. La compagnie écoule ses explosifs à partir de trois arsenaux, A, B et C. Ces arsenaux requerront respectivement, 350, 200 et 395 tonnes d'explosifs pour satisfaire la demande de leur clientèle au cours du prochain mois. Les coûts de transport des usines aux arsenaux, également assumés par Nitrobec, sont donnés au tableau 7.2.

1. rappel : toute référence doit être citée.

TABLEAU 7.2**Coûts de transport
des explosifs (en \$/t)**

Usine	Arsenal A	Arsenal B	Arsenal C
T	14	5	6
U	17	6	5

Nitrobec a dû limiter les tonnages d'explosifs qui empruntent chaque mois les routes menant des usines aux arsenaux. Ces limites, qui sont données au tableau 7.3, résultent moins de la condition des routes que des ententes conclues avec des riverains inquiets quant aux mesures prises pour assurer le transport sécuritaire des explosifs. Nitrobec envisage d'organiser subrepticement des transports d'explosifs entre certains arsenaux en empruntant des routes dont les riverains n'ont pas encore manifesté leur opposition au transit d'explosifs. On pourrait ainsi, croit-on chez Nitrobec, diminuer le total des coûts de transport. Le tableau 7.4 donne les coûts de transport pour les trajets interarsenaux envisagés. Ces coûts de transport sont symétriques; en effet, le transport d'une tonne d'explosifs est aussi coûteux à effectuer de l'arsenal i à l'arsenal j que de l'arsenal j à l'arsenal i .

TABLEAU 7.3**Limites de poids
(en t/mois)**

Usine	Arsenal A	Arsenal B	Arsenal C
T	170	180	190
U	200	150	250

TABLEAU 7.4**Coûts de transport
interarsenaux (en \$/t)**

Arsenal	A	B	C
A	—	3	—
B	3	—	2
C	—	2	—

Nitrobec doit organiser, au moindre coût, le transport de la nitrocellulose et des explosifs tout en respectant les contrats avec les fournisseurs, en n'excédant pas les limites de poids prescrites sur les routes et en satisfaisant la demande d'explosifs.

Question 2 Proposer une modélisation graphique du transport de la nitrocellulose et des explosifs pour le problème 1; décrire le graphe (sommets et arcs) de 3 manières différentes, par : sa matrice d'adjacence, sa matrice d'incidence et sa liste des successeurs;

Question 3 Quel problème classique d'optimisation combinatoire reconnaissez-vous ?

Question 4 Prouver que le problème 1 possède une solution réalisable; une solution optimale.

Question 5 Proposer une modélisation d'une instance générale du problème en un programme linéaire.

Question 6 Discuter de la nature entière/fractionnaire des solutions du problème 1.

Question 7 Soit une variante du problème, dans laquelle une tonne de nitrocellulose ne permet de produire que 0,8 tonne d'explosifs, modifier le modèle linéaire en conséquence. Modifier l'énoncé du problème pour le rendre à nouveau réalisable.

Question 8 Recenser des algorithmes applicables à la résolution de ce problème, discuter de leur complexité.

Problème 2 Nivellement de terrain. Cette application est inspirée d'un problème auquel les ingénieurs civils doivent faire face lors de la construction de routes en terrains montagneux. La figure 1 représente le tracé plan de la route à construire, à travers cols et vallées.

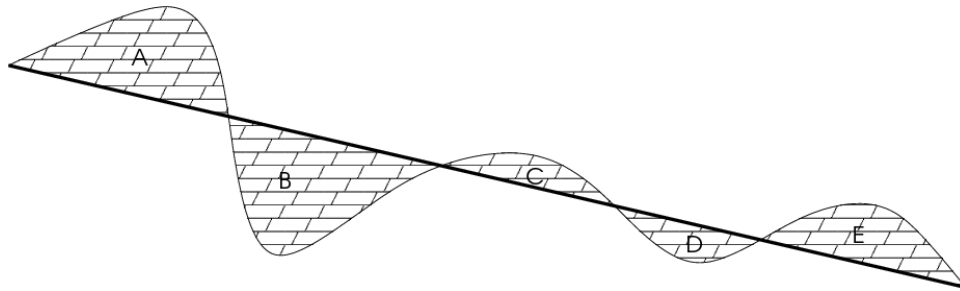


FIGURE 1 – Terrain et tracé de la route en coupe.

Il s'agit de redistribuer la terre des zones les plus élevées (A, C et E) aux zones les plus basses (B et D) afin de créer le tracé de route nivelé. On dispose pour cela d'un camion dont on cherchera à minimiser les déplacements. En considérant la charge du camion comme unité de volume, alors :

- les volumes de terre à retirer des zones A, C et E sont respectivement de 6, 3, et 5 ;
- les volumes de terre pour remblayer les zones B et D sont respectivement de 7 et 3 ;

Enfin, les distances entre les différentes zones sont de : 12km entre A et B, 15km entre B et C, 9km entre C et D, 10km entre D et E.

Question 9 *Discuter du problème 2.*

1.2 Séance 2

Problème 3 La problématique suivante concerne la protection d'informations sensibles lors de la publication de données. Soit un ensemble de données statistiques obtenues par l'INSEE à partir des réponses confidentielles d'un panel I d'individus à un ensemble J de questions. À toute question $j \in J$, la réponse d'un individu $i \in I$ consiste en une valeur x_{ij} entière positive. L'INSEE souhaite publier les moyennes de toutes ces réponses – les moyennes r_i des réponses par individu $i \in I$ et les moyennes c_j des réponses par question $j \in J$ – sans compromettre la confidentialité des réponses individuelles x_{ij} , $(i, j) \in I \times J$.

Question 10 Formuler précisément le problème de décision (instance en entrée et solution en sortie) que l'INSEE doit résoudre afin de déterminer si les moyennes sont ou non publiables.

Question 11 Exhiber une condition suffisante élémentaire, telle que la solution au problème est NON pour toute instance vérifiant la condition.

Question 12 Exprimer les formules mathématiques liant les valeurs x_{ij} , r_i , c_j , $(i, j) \in I \times J$; faire le lien avec la loi de Kirchhoff; en déduire une modélisation graphique du problème.

Question 13 Un sous-ensemble $\{(i, j, x_{ij}) \mid (i, j) \in Y \subsetneq I \times J\}$ de réponses, avec les individus et les questions qui s'y rapportent, ont été éventées. Modifier votre modèle en conséquence.

Question 14 * Soit f un (s, t) -flot compatible de valeur v dans G et G_f le graphe d'écart. Donner une condition nécessaire et suffisante sur G_f pour qu'il existe un autre (s, t) -flot compatible f' de valeur v dans G tel que $f'_{ij} = f_{ij} - 1$.² À partir de ce résultat, proposer une démarche de résolution; proposer un algorithme et donner sa complexité.

Problème 4 Grâce au tatouage de votre frère unique, vous parvenez, avec 7 de vos co-détenus, à vous évader de la prison de la Joliette. Vous devez maintenant courir pour rejoindre un aéroport voisin où un avion, prêt à décoller, vous attend. Il vous faut pour cela traverser un dédale de ruelles très étroites empêchant une fuite groupée. Malheureusement pour vous, le capitaine Bellick, qui connaît bien le plan de ces ruelles, a eu le temps de se poster le long de l'une d'entre elles de sorte à intercepter au moins l'un d'entre vous, voire plus.

Question 15 Discuter du problème 4 : modéliser la stratégie du capitaine (i.e. le problème que le capitaine doit résoudre pour savoir quelle ruelle choisir). Quelle stratégie peut vous assurer d'arriver au moins à 6 à l'aéroport? Ajouter à la modélisation. Proposer un algorithme de résolution pour le problème du capitaine; indiquer la complexité. Qualifier la topologie du réseau de ruelles, sachant qu'il ne possède ni voie souterraine, ni voie aérienne.

2. Indication : voir le problème de flot de coût minimal et sa résolution par l'algorithme dit *primal*.

1.3 Séance 3

Problème 5 En Éthiopie, suite à une récente campagne de sondages géologiques, le gouvernement fédéral (MoWR) et les bureaux régionaux des eaux (WRB) ont lancé l'exploitation de nouvelles nappes phréatiques découvertes. Les forages et les pompes mécaniques ont été installés ; il s'agit maintenant de mettre en place sur certains emplacements de forage, les équipements d'analyse et de traitement des eaux. À défaut d'être directement équipé, un forage doit nécessairement être raccordé à au moins un site de traitement : le raccordement est possible si et seulement si la distance est inférieure à 150 km. Les coûts de création et d'exploitation d'un site de traitement sont extrêmement élevés pour le pays. L'objectif est donc d'en minimiser le nombre.

Question 16 Modéliser le problème 5 en un programme linéaire en nombre entiers (P) ; réduire l'expression du modèle, en supprimant tout ensemble de contraintes induit et démontrer la validité du modèle ; comment s'appelle la matrice définissant le système linéaire de contraintes de (P) (en terme de graphe) ?

Question 17 Dualité faible/forte et bornes. Écrire le programme linéaire dual (\bar{D}) de (P). On note (D) le programme linéaire en nombres entiers dont (\bar{D}) est la relaxation continue ; on note (\bar{P}) la relaxation continue de (P). Soient x^* , \bar{x}^* , y^* et \bar{y}^* des solutions optimales de (P), (\bar{P}), (D) et (\bar{D}), respectivement. Prouver l'existence de telles solutions, puis établir une relation d'ordre entre elles. S'il existe z une solution réalisable à la fois de (P) et de (D), que dire des optima de (P) et (D) ?

Question 18 Connaissant $\Delta \in \mathbb{Z}_+$, le nombre maximal de forages voisins auxquels un forage peut être raccordé, déterminer une solution réalisable de (\bar{D}) ; en déduire une borne inférieure du problème 5 ; exprimer de manière générique³ cette borne du problème (P).

Question 19 Illustrer chacun des deux cas suivants par un exemple d'instance qui le vérifie :

1. (P) et (D) ont des valeurs optimales égales ;
2. (P), (\bar{P}) et (D) ont des valeurs optimales différentes.

Pour ces deux instances, identifier des solutions optimales de (P), (\bar{P}), (D) et (\bar{D}). Pour prouver l'optimalité des solutions de (\bar{P}) de (\bar{D}), on pourra s'appliquer d'abord à borner les valeurs optimales de ces programmes linéaires en considérant des combinaisons valides de leurs contraintes.

Question 20 Complexité. Quel est le nom usuel du problème d'optimisation modélisé par (P) ? Quelle est sa complexité théorique ? Exhiber une classe d'instances pour laquelle, le problème est trivial.

Question 21 Approximation et garantie de performance. Ajouter au glossaire les définitions de : **algorithme d'approximation** (v.s. **algorithme d'optimisation**), **ratio de performance absolu**, **algorithme r-approché**, **problème r-approximable**.

Soit $G = (V, E)$ un graphe définissant une instance quelconque de (P). Exhiber une solution réalisable de (P) de valeur maximale ; montrer que la valeur de toute solution réalisable de (P) est au plus égale à $|V|$. Soit Δ le degré maximal de G , montrer que la valeur de toute solution réalisable de (P) est au plus égale à $\Delta + 1$ fois la valeur optimale.⁴

Question 22 Proposer un algorithme glouton d'approximation pour (P) ; donner sa complexité temporelle et spatiale.

Question 23 Déterminer une solution optimale du problème 5 pour l'instance à 20 forages représentée sur la figure 2, en argumentant votre raisonnement.

Question 24 Votre solution, répondant à la question précédente, a été retenue et les usines de traitement ont été construites aux emplacement préconisés. La seconde étape consiste à réaliser un réseau de canalisation de longueur totale minimale permettant de relier l'ensemble des 20 forages. Ce réseau est composé de deux types de canalisations : des raccords forage-usine (dont la longueur est non nulle et inférieure à 150 km), et des raccords usine-usine. Quelles sont les caractéristiques d'un tel réseau ? Comment appelle-t-on un tel objet en terme de graphe ? Proposer un algorithme de construction du réseau.

3. comment appelle-t-on Δ en terme de graphes ?

4. Hint : borner par une combinaison des contraintes de (P).

Figure 4 Status of hydrogeological mapping in Ethiopia, 2004

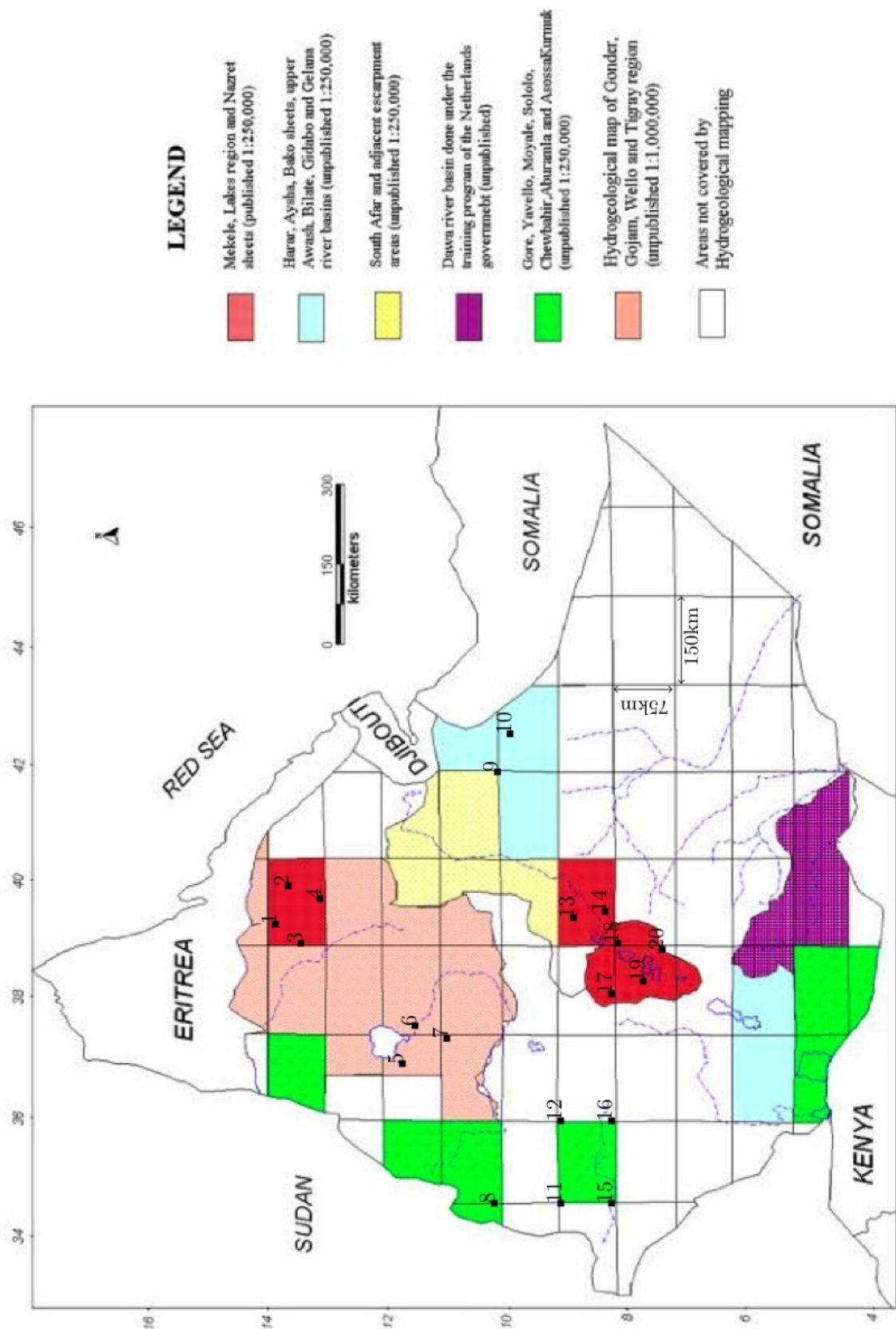


FIGURE 2 – instance à 20 forages du problème 5

Dans la suite, on considèrera un graphe $G = (V, E)$ non-orienté. Pour simplifier la notation, on posera $E \subseteq V \times V$ en assimilant $(i, j) = (j, i)$ pour désigner une même arête $\{i, j\}$.

Question 25 Ajouter au glossaire, la définition et le nom usuel français de chacun des problèmes suivants :

minimum edge cover

minimum node cover

maximum clique

maximum matching

maximum independant set

maximum stable

maximal matching

maximal independant set

Problème 6 Accès wifi. Une municipalité a décidé d'offrir un accès wifi à tous les habitants du centre-ville. Elle veut équiper le plus petit nombre possible de carrefours en bornes wifi (dont la longueur d'émission est supérieure à la plus longue rue de la ville) de manière à ce que, depuis chaque habitation dans chaque rue, au moins un signal soit capté.

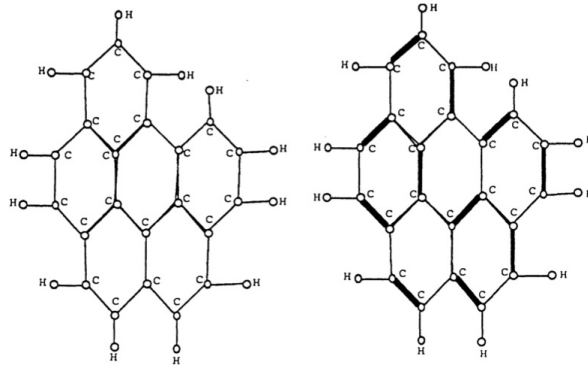
Problème 7 Reconnaissance d'images. Soient deux images A et B de taille identique, on s'intéresse à la proximité de ces deux images selon une mesure simple⁵ : la fréquence d'apparition de couleurs. Par exemple, selon une telle mesure, deux images bichromes, l'une composée de 200 pixels rouges et de 56 pixels jaunes, l'autre composée de 206 pixels noirs et de 50 pixels blancs, sont considérées comme quasi-identiques : en appariant 200 pixels rouges/noirs et 50 pixels jaunes/blanc, on obtient une « distance » égale à 250/256. De manière générale, on apparie tous pixels de A à tous pixels de B et on cherche le nombre maximal de couples mutuellement cohérents (divisé par la taille des images) : deux couples $(a_1, b_1) \in A \times B$ et $(a_2, b_2) \in A \times B$ sont dits cohérents si et seulement si a_1 et a_2 sont de la même couleur et b_1 et b_2 sont de la même couleur, ou bien, a_1 et a_2 sont de couleurs différentes et b_1 et b_2 sont de couleurs différentes.

Problème 8 Ordonnancement de tâches sur bi-processeur. Étant donnés deux processeurs, un ensemble de tâches de durée unitaire et une relation de compatibilité entre paires de tâches, l'ordonnanceur du système doit déterminer le temps minimal nécessaire à l'exécution de l'ensemble des tâches au moyen des deux processeurs, de sorte que : (i) une tâche est exécutée par l'un ou l'autre des deux processeurs (ii) un processeur exécute une tâche à la fois (iii) deux tâches exécutées en parallèle sur les deux processeurs sont nécessairement compatibles.

Problème 9 Débat politique. Une émission radiophonique new-yorkaise souhaite organiser un débat politique entre partisans démocrates et républicains. Les participants ont déjà été sélectionnés, et les échanges s'annoncent houleux. Afin d'éviter toute querelle familiale, les producteurs décident, en dernière minute, de ne pas faire participer des membres d'une même famille qui seraient de tendances politiques opposées. Le nombre de participants doit cependant rester le plus large possible.

Problème 10 Structure moléculaire. Une molécule est définie par la liste et le nombre de ces atomes, mais aussi par sa structure : l'ordre des liaisons inter-atomiques. À partir de la première information, les chimistes ont donc à déterminer la structure atomique possible d'une molécule, c'est à dire affecter un ordre (simple, double, etc.) aux liaisons inter-atomiques de sorte que la valence des atomes soit respectée. On considère une molécule hydrocarbonée, composée d'atomes C de carbone (de valence 4) et H d'hydrogène (de valence 1). Les liaisons $C - H$ sont nécessairement simples mais les liaisons $C - C$ peuvent être simples ou doubles. Avec des liaisons simples (à gauche sur le schéma), les valences des atomes de carbone C seraient toutes égales à 3. Il s'agit donc de déterminer quelles liaisons $C - C$ doubler pour que chaque atome de carbone ait une valence de 4.

5. L'application est ici simplifiée, mais le raisonnement est identique pour des mesures plus complexes.



Question 26 Qualifier les problèmes 6 à 10.

Question 27 Qualifier les deux problèmes modélisés par les programmes linéaires en nombres entiers suivants :

$$(P_x) : \quad \max \left\{ \sum_{e \in E} x_e \mid \sum_{e \in E | i \in e} x_e \leq 1 \ (\forall i \in V), \ x \in \mathbb{Z}_+^E \right\} \quad (1)$$

$$(P_u) : \quad \max \left\{ \sum_{i \in V} u_i \mid u_i + u_j \leq 1 \ (\forall (i, j) \in V^2 \setminus E), \ u \in \mathbb{Z}_+^V \right\} \quad (2)$$

Trouver leurs classes de complexité respectives.

Question 28 Modéliser les 6 premiers problèmes de la question 25 en programmes linéaires en nombres entiers. Exhiber un maximum de relations entre ces différents problèmes (par complémentarité, dualité, etc.). En déduire, la classe de complexité de tous.

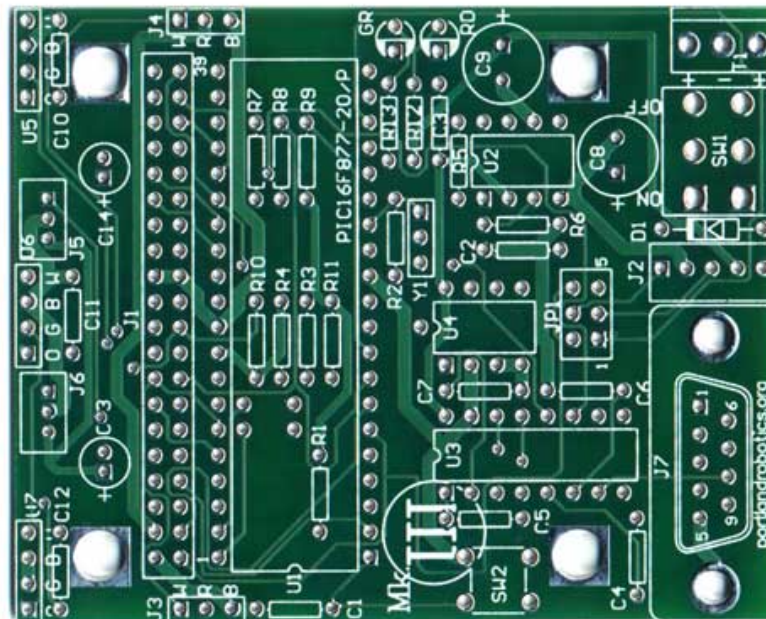
Question 29 Identifier, parmi les problèmes 6 à 10, ceux résolubles en temps polynomial.

Question 30 Approximation pour la couverture de sommets. Proposer un algorithme glouton d'approximation pour le problème du transversal de cardinalité minimale, où, à chaque itération, un sommet est ajouté au transversal en construction. On trouvera dans la littérature, la preuve qu'un tel algorithme est $\ln |V|$ -approché. Étant donné un couplage maximal $X \subseteq E$, montrer que $S = \{i \in V \mid \exists e \in X \wedge i \in e\}$ est un transversal de cardinalité inférieure ou égale à 2 fois la cardinalité minimale d'un transversal ; proposer alors un second algorithme basé sur la recherche glouton d'un couplage maximal. Donner et commenter la garantie de performance de ce second algorithme.

2 Seconde partie : Tournées

2.1 Séance 4

Problème 11 Une des premières étapes dans la fabrication de circuits imprimés consiste à percer la plaque support de silice aux endroits définis pour fixer les composants électroniques. En entrée, on dispose d'une plaque sur laquelle sont marqués les emplacements des trous à faire au moyen d'une perceuse automatique mobile. La perceuse peut réaliser les trous dans n'importe quel ordre, en se déplaçant sur un plan horizontal au-dessus de la plaque, et sa vitesse de déplacement est constante. Les positions initiale et finale de la perceuse sont au-dessus du centre de la plaque. On souhaite déterminer l'ordre de perçage des trous qui minimise le déplacement total de la machine.



Question 31 Modéliser cette problématique en une instance du problème du voyageur de commerce (Traveling Salesman Problem – TSP). Qualifier cette instance du TSP (symétrique, asymétrique, euclidien,...) ? Étant donnée une solution optimale du problème – on note π_i le i -ème trou percé suivant cette solution – exprimer la valeur optimale du TSP équivalent.

Le problème d'optimisation considéré ci-après est le **TSP euclidien** sur un ensemble V de villes : trouver la plus courte route non-orientée qui passe par chacune des villes et qui termine là où elle commence. On note $n = |V|$ le nombre de villes et $G = (\mathbb{K}^V, c)$ le graphe complet non-orienté sur l'ensemble V des sommets et pondéré par la matrice c des distances entre les villes.

Question 32 Caractérisation des solutions. Montrer que dans toute solution optimale du TSP euclidien, chaque ville est visitée exactement une fois. Montrer que le problème est équivalent à trouver un cycle (simple) hamiltonien de G de plus petite longueur. Donner deux représentations/encodages d'un cycle hamiltonien dans G – par une séquence de sommets ou par un ensemble d'arêtes – en précisant la taille de ces ensembles et les contraintes définissant ces ensembles. Combien de cycles hamiltoniens G possède-t-il ? Par la suite, on désignera par **tour** un cycle hamiltonien dans G et par **tour partiel** un cycle dans G .

Construction de tours.

Question 33 On s'intéresse à des heuristiques constructives gloutonnes de tours : partant d'un unique sommet ou, plus généralement, d'un tour partiel, la construction consiste à augmenter le tour partiel courant d'un nouveau sommet à chaque itération, jusqu'à ce que tous les sommets soient ajoutés. Le choix du tour partiel initial

est défini par la **politique d'initialisation** de l'heuristique ; Le choix du prochain sommet ajouté à chaque itération est défini par la **politique de sélection** ; le choix du point d'insertion de ce nouveau sommet est défini par la **politique d'insertion**. Écrire les algorithmes des heuristiques gloutonnes correspondant aux politiques de sélection/insertion suivantes :

voisin le plus proche : ajoute, à la suite du dernier sommet du tour partiel, son voisin non encore visité le plus proche ;

insertion la plus proche : sélectionne un sommet non visité arbitrairement et l'ajoute au tour partiel juste après le sommet le plus proche ;

insertion la moins coûteuse : sélectionne un sommet non visité arbitrairement et l'ajoute au tour partiel à l'endroit qui résulte en l'augmentation minimale du coût du tour partiel.

Quel est le défaut principal de la première approche en comparaison des deux autres ? Discuter de la complexité et des structures de données les plus adaptées à ces algorithmes.

Question 34 Modifier les deux derniers algorithmes pour utiliser la politique de sélection suivante :

sélection du plus proche : sélectionne le sommet non visité le plus proche de l'ensemble des sommets du tour partiel.

Comparer l'augmentation du coût du tour partiel obtenu à chaque itération pour chacun des 3 algorithmes.

Question 35 Modifier le dernier algorithme pour utiliser les politiques d'initialisation et de sélection suivantes :

initialisation : le tour partiel initial est formé des deux sommets les plus distants ;

sélection du plus distant : sélectionne à chaque itération le sommet non visité le plus distant de l'ensemble des sommets du tour partiel.

Discuter l'intérêt de cette démarche en illustrant, par exemple, le résultat sur une instance.

Question 36 Soit l'algorithme constructif suivant :

input : $G = (\mathbb{K}^V, c)$ graphe complet non-orienté sur V muni de la distance eulérienne c .
step 0. $\mathcal{T} = \{ (i) \mid i \in V \}$
step 1. si $|\mathcal{T}| = 1$, **STOP RETURN** $T \in \mathcal{T}$
 sinon choisis $T \neq T'$ dans \mathcal{T} tels que $\min\{c_{ij} \mid i \in T, j \in T'\}$ soit minimal
step 2. si $T = (i)$ et $T' = (j)$ alors $T'' = (ij)$
 sinon si $T = (i)$ et $T' = (j_1 j_2 \dots j_k)$ alors $T'' = (ij_1 j_2 \dots j_k)$
 sinon si $T = (j_1 j_2 \dots j_k)$ et $T' = (i)$ alors $T'' = (ij_1 j_2 \dots j_k)$
 sinon si $T' = (i_1 i_2 \dots i_l)$ et $T = (j_1 j_2 \dots j_k)$ alors $T'' = (i_1 i_2 \dots i_l j_1 j_2 \dots j_k)$
step 3. $\mathcal{T} = \mathcal{T} \setminus \{T, T'\} \cup \{T''\}$, **GOTO step 1**

où $(i_1 i_2 \dots i_l)$ est la notation usuelle d'une permutation d'éléments. Quel est la valeur de retour de cet algorithme ? En décrire chaque étape. Proposer une amélioration de **step 2**.

Question 37 Quelle relation existe-t'il entre le problème de l'**arbre de recouvrement de poids minimal** dans un graphe et le TSP ? Citer deux célèbres algorithmes de calcul de l'arbre de recouvrement de poids minimal dans un graphe et donner leur complexité. Faire le parallèle entre ces algorithmes et les algorithmes des questions 34 et 36. Citer une autre heuristique pour le TSP basée sur la recherche d'un arbre de recouvrement de poids minimal du graphe.

Amélioration de tours. Une procédure de **recherche locale** consiste à partir d'une solution initiale et lui appliquer une série de transformations pour aboutir à une solution de meilleur coût. Une telle procédure repose sur quatre éléments : la construction d'une solution initiale, la liste de transformations applicables, la politique de sélection de la transformation à appliquer à chaque itération, et la condition d'arrêt.

Une condition d'arrêt communément utilisée est l'obtention d'un **optimum local**, c'est à dire une solution telle qu'aucune transformation autorisée ne mène à une solution meilleure. Une condition de sélection de la transformation à appliquer à chaque itération est que celle-ci améliore la solution courante. Sous ces deux conditions, et si le problème est borné, la procédure est assurée de terminer. La **méthode de descente** (*hill-climbing*) est un exemple de procédure de recherche locale qui sélectionne à chaque itération, la transformation menant à la meilleure solution atteignable.

On a traité de la construction de solutions initiales pour le TSP dans les questions précédentes⁶. On considère maintenant des classes de transformations locales possibles. Un **mouvement** de recherche locale est une opération qui transforme une solution réalisable en une autre solution réalisable. Le **voisinage** d'une solution associé à un mouvement est l'image de la solution par cette opération.

Question 38 *Montrer que les opérations suivantes sont des mouvements de recherche locale, les décrire formellement ou les illustrer, et donner la taille d'un voisinage :*

inversion de sommets : échange deux sommets consécutifs du tour ;

échange de sommets : échange deux sommets distincts du tour ;

échange d'arêtes 2-opt : supprime deux arêtes non-adjacentes du tour et reconnecte (une seule possibilité) les deux chemins résultant.

Question 39 *Illustrer les deux mouvements suivants :*

échange d'arêtes 3-opt : supprime trois arêtes du tour et reconnecte les chemins résultants.

échange d'arêtes Or-opt : supprime trois arêtes telles que l'un des chemins résultant contienne exactement 1, 2 ou 3 sommets et reconnecte les trois chemins résultants.

Déterminer l'ordre de grandeur des voisinages correspondant. Montrer que les voisinages 2-opt et Or-opt sont inclus dans le voisinage 3-opt et qu'un mouvement 3-opt est équivalent à au plus deux mouvements 2-opt successifs. Remarquer que ces transformations se généralisent au mouvement r -opt, pour $r \geq 1$, et discuter de l'applicabilité de ce type de voisinage.

Pour en savoir plus :

- la procédure *Or-opt* [4] est une recherche locale où à chaque itération, sont appliqués tous les mouvements Or-opt qui diminuent le coût de la solution courante : d'abord les mouvements de réinsertion de 2 villes puis de 1 ville. C'est une procédure simple à implémenter, relativement peu coûteuse et qui donne de bons résultats ;
- la procédure *variable r-opt* de Lin et Kernighan [3] est une recherche locale qui à chaque itération effectue un mouvement r -opt avec r ajustable : plus précisément, à chaque itération, une série potentiellement longue de mouvements 2-opt est appliquée de sorte que chaque sous-série initiale de mouvements mène à une meilleure solution, mais qu'en revanche, un de ces mouvements 2-opt isolé peut causer l'augmentation du coût. Cette procédure est difficile à implémenter mais elle est très performante et l'une des meilleures à ce jour ;
- les politiques de sélection de mouvement et conditions d'arrêt décrites dans les questions précédentes assurent la terminaison de la procédure de recherche locale, mais celle-ci aboutit alors en un optimum local sans possibilité d'en sortir (pour atteindre un optimum global). Une technique souvent utilisée est de relancer alors la procédure si celle-ci n'est pas entièrement déterministe (en partant d'une solution initiale différente ou en insérant des choix aléatoires dans la procédure). On retient alors le meilleur optimum local atteint parmi toutes les relances. L'approche alternative permettant d'échapper aux optima locaux est de considérer des périodes d'**intensification** de la recherche vers des solutions de meilleurs coût (comme dans une recherche locale, en considérant également des critères probabilistes dans la politique de sélection) et des périodes de **diversification** où l'on va appliquer des transformations plus globales de la (ou des) solution(s) courante(s)

6. en partie du moins : il existe beaucoup d'autres heuristiques constructives pour le TSP

de manière à aller explorer d'autres parties de l'espace de recherche. Cette approche est celle des **metaheuristiques** telles que les méthodes du **recuit simulé**, la **méthode tabou**, les **algorithmes génétiques**, les **colonies de fourmis**, etc.

Relaxations, bornes inférieures et branchement

Question 40 Montrer que le coût minimal d'un arbre de recouvrement de G est au plus égal à l'optimum du TSP. Proposer une borne inférieure du TSP basée sur le calcul d'un arbre de recouvrement de poids minimal. Quel autre problème classique de graphe forme aussi une relaxation du TSP. Avec quelles complexités les bornes inférieures issues de ces relaxations peuvent-elles être calculées ?

Question 41 Soit les variables de décision $y_k \in V, 1 \leq k \leq ?$ définies par $y_k = i$ si et seulement si i est la k -ème ville visitée par le tour. Décrire une méthode exacte de recherche arborescente pour le TSP basée sur le branchement sur ces variables. A minima la méthode devra couper tout nœud de l'arbre de recherche dont l'instanciation partielle correspondante des variables de décision ne constitue pas un tour partiel. Le branchement pourra également assurer qu'un tour ne soit présent qu'à au plus une feuille de l'arbre de recherche. Enfin, on pourra ajouter à la méthode des calculs de bornes (incrémentales si possible).

Question 42 Même question avec les variables de décision $x_e \in \{0, 1\}$ définies par $x_e = 1$ si et seulement si l'arête e de G appartient au tour. Même question avec les variables de décision $z_i \in V$ définies par $z_i = j$ si et seulement si la ville j suit la ville i dans le tour.

2.2 Séance 5

Problème 12 Les chaînes de construction et de montage des voitures fourmillent de problèmes d'optimisation combinatoire. L'un des problèmes qui se posent consiste à ordonner le passage d'un lot de voitures sur la chaîne de peinture de sorte que la durée totale de passage soit minimale. On connaît la couleur attribuée à chacune des voitures du lot ainsi que son temps de passage sur la machine (qui dépend du niveau d'équipement de la voiture et de la couleur). La chaîne de peinture peut opérer sur toutes les voitures, quelque soit la couleur et quelque soit l'ordre. Cependant, le temps de préparation de la machine entre deux voitures dépend des couleurs de ces voitures : ce laps de temps, qui est connu précisément, est plus ou moins long selon si les deux voitures sont de même couleur (léger nettoyage des jets), ou bien si la seconde voiture est de teinte foncée (nettoyage moyen et changement de couleur) ou bien si la seconde voiture est de teinte claire (nettoyage supérieur et changement de couleur). Enfin la chaîne est nettoyée en profondeur après chaque passage de lot, ce qui induit un temps final plus long encore, qui dépend aussi de la dernière couleur utilisée.



Question 43 Identifier toutes les données d'entrée, puis modéliser le problème.

On reprend ici les notations de la section précédente : soit un graphe complet non-orienté pondéré $G = (K^V, c)$ à $n = |V|$ sommets.

Question 44 assignment problem. On considère le **problème d'affectation linéaire** suivant : trouver une fonction bijective f de V (ensemble fini discret) dans V telle que $\sum_{i \in V} c_i f(i)$ soit minimal. Formuler en un programme linéaire en variables booléennes (PLVB). À quelle classe de complexité ce problème appartient-il ? Quelle relation existe-t'il entre ce problème et le TSP ? Proposer une seconde formulation de PLVB pour le problème d'affectation linéaire avec exactement $|V|$ contraintes.

Question 45 subtour cuts. Soit W un sous-ensemble strict non-vide de V ($\emptyset \subsetneq W \subsetneq V$) et A l'ensemble des arêtes de G ayant exactement une extrémité dans W . Un tour (i.e. un cycle hamiltonien) T de G ne peut être entièrement inclu dans le sous-graphe de G engendré par W . Quelle relation existe-t'il entre T et le graphe partiel engendré par A ? Modéliser cette condition au moyen des variables de la question précédente : une telle contrainte est appelée contrainte d'élimination de sous-tour. En déduire une formulation linéaire du TSP ; évaluer sa taille.

Question 46 control zones. On considère ici un encodage du TSP, différent encore, basé sur exactement n disques, chacun centré sur un sommet de G , et ne s'intersectant pas (les disques sont soit deux à deux disjoints, soit seulement adjacents). Comparer la somme des diamètres des disques et le coût d'un tour dans G . Proposer un programme linéaire modélisant une relaxation du TSP et basé sur des variables continues $r_i > 0$: pour tout sommet $i \in V$, r_i est la longueur du rayon du disque centré en i . Considérer le dual de ce PL.

Question 47 moats. Montrer sur un exemple, que l'optimum du PL de la question précédente n'atteint pas nécessairement l'optimum du TSP. Soit une solution optimale (r_1, \dots, r_n) du PL et un sous-ensemble $W \subsetneq V$ tel qu'aucun disque centré sur un sommet de W n'est adjacent à un disque centré sur un sommet de $V \setminus W$. Exprimer la valeur maximale $u_W > 0$ telle que, pour tous $i \in W$ et $j \in V \setminus W$, le disque centré sur i de rayon $r_i + u_W$ n'intersecte pas le disque centré sur j de rayon r_j . Que déduire de cette information pour le TSP ? Proposer un algorithme glouton d'amélioration de la borne du PL.

Question 48 Proposer un modèle linéaire du problème de cycle hamiltonien (on prendra une fonction objective constante quelconque) basé sur des variables booléennes de type $y_{ik} = 1$ si et seulement si i est le k -ème sommet du cycle. Compléter ce modèle en un PLVB du TSP de taille polynomiale en n .

Dantzig, Fulkerson et Jonhson [2] furent les premiers à considérer la relaxation AP et les contraintes de sous-tours pour le TSP. Partant de la solution optimale de (P) le PLVB de la question 44, ils proposèrent la méthode de résolution itérative suivante : à chaque itération, la relaxation continue de (P) est résolue et une contrainte de sous-tour, violée par la solution courante, est recherchée puis, si elle existe, ajoutée à (P). Cette méthode est aujourd'hui appelée *algorithme de génération de coupes*.

Pour les questions de recherche bibliographique suivantes, on pourra s'appuyer notamment sur les documents disponibles sur [6] et qui reprennent en partie le livre [1].

Question 49 Montrer que l'algorithme termine. Que peut-on dire de la solution retournée ? Définir les méthodes de séparation et de branch-and-cut.

Question 50 Résumer la méthode de [2] de recherche de coupes de sous-tour. S'agit-il d'une heuristique ou, au contraire, la méthode assure-t-elle de trouver une coupe tant qu'il en existe ?

- citer d'autres méthodes de recherche de coupes de sous-tour ;
- citer d'autres classes de coupes structurelles propres au TSP ;
- citer une classe de coupes générique à tout PLNE.

2.3 Séance 6 – rédacteurs : Fabien Le Huédé et Sophie Demasse

2.4 Problem statement

The Vehicle Routing Problem with Time Windows (VRPTW) is the extension of the Capacitated VRP where the service at each customer i must start within a time interval $[a_i, b_i]$ called a time window. During the service time, denoted s_i , the vehicle stays at the location of customer i . If it arrives at this location before time a_i , it has to wait until the customer is ready to start service.

The VRPTW consists in finding a set of k_{\max} routes, one for each vehicle, such that :

1. each vehicle starts from and returns to one specific location called the depot,
2. each customer is visited by exactly one vehicle,
3. the sum of the demands of the customers visited by a vehicle does not exceed its capacity C ,
4. for each customer i , the service starts within the time window $[a_i, b_i]$ and it lasts s_i time units,
5. the travel between two locations i and j – when it is allowed – lasts t_{ij} time units and it has a financial cost c_{ij} ,
6. the sum of the vehicle travel costs is minimized.

2.5 Model

Let $N = \{1, \dots, n\}$ be the set of customers and $K = \{1, \dots, k_{\max}\}$ be the set of vehicles that can be used to perform the service. The VRPTW is defined on a network $G = (V, A)$, where $V = \{0, \dots, n+1\}$ and the nodes 0 and $n+1$ represent the depot : all feasible vehicle routes correspond to elementary paths from node 0 to node $n+1$ in G . For each arc $(i, j) \in A$, we denote t_{ij} the travel time on this arc and c_{ij} the travel cost. For each vertex $i \in V$, we denote $\delta^+(i)$ the set of successors of i and $\delta^-(i)$ the set of predecessors of i .

Nodes 0 and $n+1$ are given a time window $[E, L]$ where E and L represent the earliest departure from the depot and the latest possible arrival at the depot, respectively. In addition, zero demands and service times are defined for these nodes ($d_0 = d_{n+1} = s_0 = s_{n+1} = 0$). Unused vehicles can be represented by routes having exactly one arc : the arc $(0, n+1)$ defined with $t_{0,n+1} = c_{0,n+1} = 0$.

The following mathematical model uses two types of variables : flow variables x_{ij}^k , $(i, j) \in A$, $k \in K$, equal to 1 if arc (i, j) is used by vehicle k and 0 otherwise, and time variables w_i^k , $i \in V$, $k \in K$, specifying the start of service at node i when serviced by vehicle k .

$$\min z = \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} x_{ij}^k \quad (3)$$

$$\text{s.t. } \sum_{k \in K} \sum_{j \in \delta^+(i)} x_{ij}^k = 1 \quad \forall i \in N, \quad (4)$$

$$\sum_{i \in \delta^-(j)} x_{ij}^k - \sum_{i \in \delta^+(j)} x_{ji}^k = 0 \quad \forall k \in K, j \in N, \quad (5)$$

$$\sum_{i \in \delta^+(0)} x_{0i}^k = 1 \quad \forall k \in K, \quad (6)$$

$$\sum_{i \in \delta^-(n+1)} x_{i(n+1)}^k = 1 \quad \forall k \in K, \quad (7)$$

$$x_{ij}^k (w_i^k + s_i + t_{ij} - w_j^k) \leq 0 \quad \forall k \in K, (i, j) \in A, \quad (8)$$

$$a_i \sum_{j \in \delta^+(i)} x_{ij}^k \leq w_i^k \leq b_i \sum_{j \in \delta^+(i)} x_{ij}^k \quad \forall i \in N, \forall k \in K, \quad (9)$$

$$E \leq w_i^k \leq L \quad \forall k \in K, i \in \{0, n+1\}, \quad (10)$$

$$\sum_{i \in N} d_i \left(\sum_{j \in \delta^+(i)} x_{ij}^k \right) \leq C \quad \forall k \in K, \quad (11)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall k \in K, (i, j) \in A. \quad (12)$$

Question 51 model analysis.

1. Explain each constraint of the model.
2. Explain why subtour elimination is ensured by this model.
3. Propose a linear formulation for constraint (8).⁷
4. In order to reduce the graph (and the number of variables) as a preprocessing step, consider temporal or capacity limitation arguments to eliminate arcs that do not enter in any solution.
5. Is it useful to distinguish between the various vehicles in this model? Propose a second model where the vehicles are not distincts. Consider the objective of minimizing the number of vehicles used and propose a linear formulation of this objective in both models.

2.6 Local search algorithm for VRPTW

This part relies on the study of Savelsbergh's paper [5]. These results are given for the TSP, so they can be used in the VRP for routes improvements. The following questions deal with algorithms proposed for the 2-opt (2-interchange) neighborhood only.

Question 52 The lexicographic search algorithm.

1. Relate Savelsbergh notations to the former notations. Explain the differences.
2. Could we suppose that service time is included in transportation times?
3. In pseudo-code, write the lexicographic search algorithm for 2-Opt explained in the paper.
4. Summarize the acceptability conditions of a 2-opt move.
5. Consider problem P_1 of Figure 3 and the route described in Figure 4, build the tours following the lexicographic search for $i = 0$ and $i = 1$. Note : the graph is complete although only selected edges are represented on the figure. Time windows are given next to each vertex on the graph.

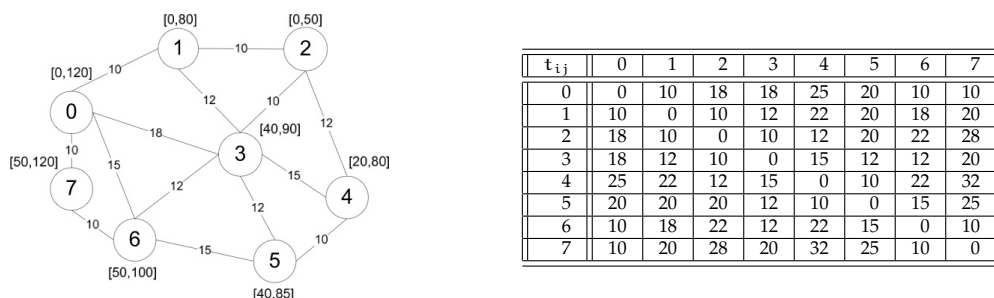


FIGURE 3 – A graph description and table of times for P_1

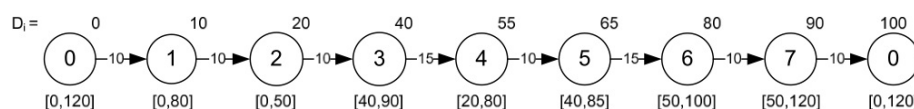


FIGURE 4 – Feasible solution for P_1

Question 53 Incremental version of the algorithm. Savelsbergh proposes an efficient implementation of the lexicographic search algorithm, using additional global variables to reduce the checking effort to $O(1)$ time for each 2-exchange.

1. Summarize the initialization and the update steps for these variables.
2. Run the algorithm at iteration $i = 2$ using this technique.
3. At the end of page 293, the author quickly describe a profitability test but does not explain it. Propose a method for this test and give its complexity.

7. Hint : consider $x_{ij}^k = 1$ then introduce an arbitrary large constant M s.t. it remains unconstrained when $x_{ij}^k = 0$.