9. Tournées

20 Décembre 2023

Problèmes de tournées

Grossièrement on peut subdiviser les problèmes de tournées en deux catégories.

1. on veut parcourir tous les sommets : c'est le problème du voyageur de commerce ;
2. on veut parcourir toutes les "arêtes" : c'est le problème du postier.

De plus on peut décliner les problèmes selon que l'on a affaire à un graphe orienté ou à un graphe non orienté.

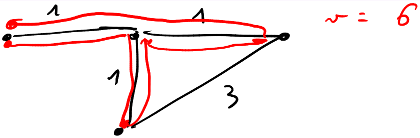
Lequel est le plus difficile ?

Problème du voyageur de commerce

Problème du voyageur de commerce (TSP)

- **Instance**. Graphe $G = (V, E)$, fonction de coût $d : E \to \mathbb{R}_+$.
- **Question**. Trouver une chaîne fermée C passant par tous les sommets telle que $\sum_{e \in C} d(e)$ minimum.

Formulation équivalente



On peut vérifier que ce problème est équivalent à :

Problème du voyageur de commerce

- **Instance**. Graphe complet $K_n = (V, E)$, fonction de coût $c : E \to \mathbb{R}_+$ telle que $c(uv) + c(vw) \geq c(uw)$ pour tous sommets u, v, w.
- **Question**. Trouver un cycle hamiltonien C passant par tous les sommets tel que $\sum_{e \in C} c(e)$ minimum.

Pourquoi ?

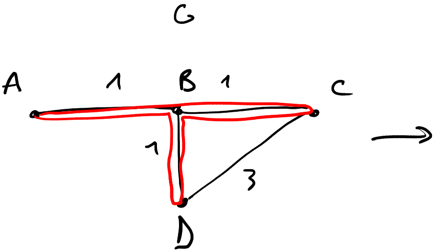
Formulation équivalente

On peut vérifier que ce problème est équivalent à :

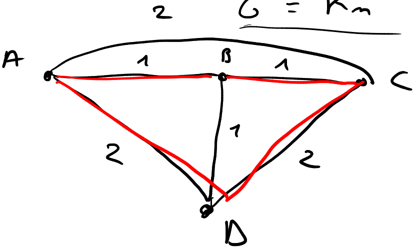
Problème du voyageur de commerce

- **Instance**. Graphe complet $K_n = (V, E)$, fonction de coût $c : E \to \mathbb{R}_+$ telle que $c(uv) + c(vw) \geq c(uw)$ pour tous sommets u, v, w.
- **Question**. Trouver un cycle hamiltonien C passant par tous les sommets tel que $\sum_{e \in C} c(e)$ minimum.

Pourquoi ?

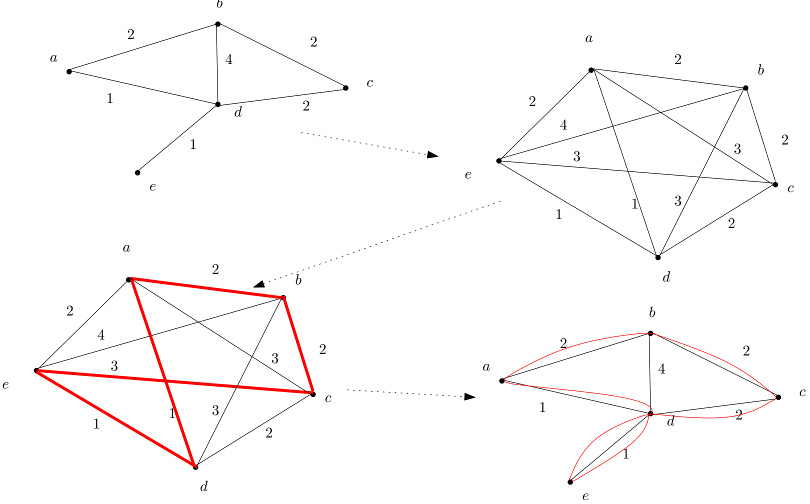


$\underline{G' = K_n}$



Reformulation

Graphe complet des plus courtes chaînes



Complexité

Theorem

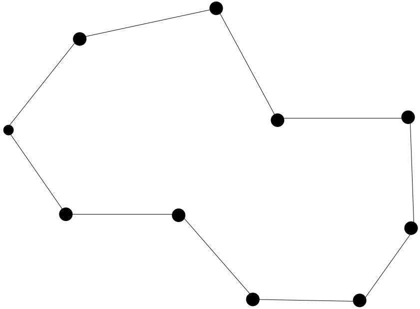
Le problème du voyageur de commerce est **NP***-difficile.*

Metric TSP

Metric TSP : $c(uv)$ is the euclidean distance.

Metric TSP

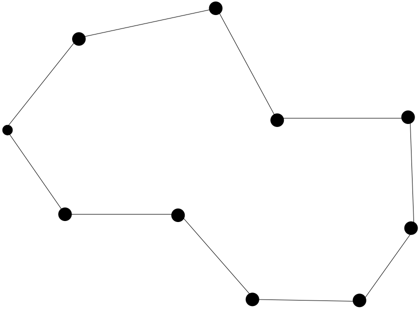
Metric TSP : $c(uv)$ is the euclidean distance.



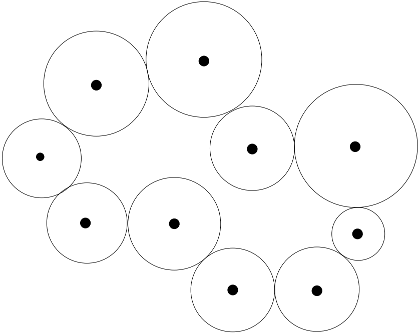
Explain why this tour is optimal.

Metric TSP

Metric TSP : $c(uv)$ is the euclidean distance.



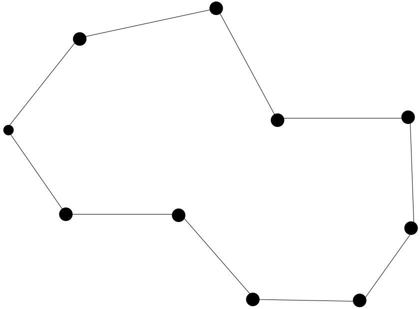
Explain why this tour is optimal.



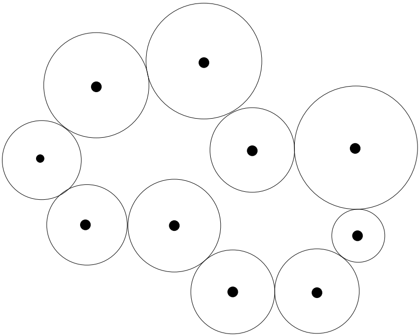
In any tour, the distance from v to its neighbors is non-smaller

Metric TSP

Metric TSP : $c(uv)$ is the euclidean distance.



Explain why this tour is optimal.



In any tour, the distance from v to its neighbors is non-smaller

Metric-TSP is NP-hard

Description linéaire des cycles hamiltoniens

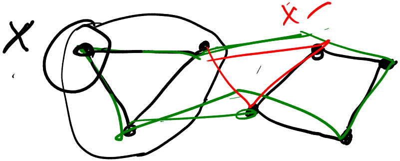
$G = (V, E)$ $\quad x_e \in \{0, 1\}$. $\quad \min_x \sum_{e \in E} c_e x_e$

Donner un polyhèdre dont les solutions entières sont les cycles Hamiltoniens.

- $\sum_{e \in \delta(v)} x_e = 2$, $\quad \forall v \in V \rightarrow$ cycle

$|V|$ contraintes

n'empêche pas d'avoir des sous-tours.



$X \subsetneq V$

$\sum_{e \in \delta(X)} x_e \geq 2$, $\quad \forall \emptyset \subsetneq X \subsetneq V \mid$ $2^n - 2$ contraintes

$\rightarrow$ nb exponentiel de contraintes

Description linéaire des cycles hamiltoniens

Donner un polyhèdre dont les solutions entières sont les cycles Hamiltoniens.

Proposition. Les vecteurs d'incidence des cycles hamiltoniens sont exactement les vecteurs solutions de

$$\begin{aligned} & x_e \in \{0,1\} && \forall e \in E, \\ & \sum_{e \in \delta(v)} x_e = 2 && \forall v \in V, \\ & \sum_{e \in \delta(X)} x_e \geq 2 && \forall X \subseteq V,\ X \neq \emptyset, V. \end{aligned}$$

Formulation du TSP comme programme linéaire

Résoudre le problème du voyageur de commerce, c'est donc résoudre le programme

$$\begin{aligned} \min_{x} \quad & \sum_{e\in E} c(e)x_e \\ \text{s.c.} \quad & \sum_{e\in\delta(v)} x_e = 2 \quad \forall v \in V, \\ & \sum_{e\in\delta(X)} x_e \geq 2 \quad \forall X \in \mathcal{C} \\ & x_e \in \{0,1\} \quad \forall e \in E, \end{aligned} \qquad \text{(P)}$$

où $\mathcal{C} = \{X \subseteq V : X \neq \emptyset, V\}$.

Peut-on résoudre ce PLNE frontalement avec un solveur ?

Formulation du TSP comme programme linéaire

Résoudre le problème du voyageur de commerce, c'est donc résoudre le programme

$$\begin{aligned} \min_x \quad & \sum_{e\in E} c(e)x_e \\ \text{s.c.} \quad & \sum_{e\in\delta(v)} x_e = 2 \quad \forall v \in V, \\ & \sum_{e\in\delta(X)} x_e \geq 2 \quad \forall X \in \mathcal{C} \\ & x_e \in \{0,1\} \qquad \forall e \in E, \end{aligned} \tag{P}$$

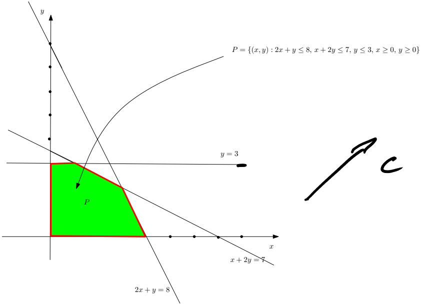
où $\mathcal{C} = \{X \subseteq V : \; X \neq \emptyset, V\}$.

Peut-on résoudre ce PLNE frontalement avec un solveur ?

Nombre de contraintes exponentiel

Génération de ligne en PL : intuition géométrique

Seulement les contraintes définissant le sommet de la solution optimale sont requises en programmation linéaire.



Quel algorithme proposez vous pour résoudre un PL avec un nombre exponentiel de contraintes ?

Génération de lignes.

(variables continues)

Génération de lignes pour un programme linéaire (P) avec un ensemble $\mathcal{C}$ de taille exponentielle.

1. Initialiser $\widetilde{\mathcal{C}} \subseteq \mathcal{C}$ d'une taille tractable.
2. Résoudre (P) avec uniquement les contraintes $\widetilde{\mathcal{C}}$
3. Identifier la contrainte la plus violée
   - S'il existe une telle contrainte, l'ajouter $\widetilde{\mathcal{C}}$
   - Sinon, on a la solution optimale

Le problème d'identification de la contrainte la plus violée est appelé problème de *séparation* (pricing en anglais)

Séparer les contraintes

Idée : on suppose que l'on dispose d'un algorithme $\mathcal{A}$ qui résout

Séparation des contraintes pour TSP

- **Instance**. Vecteur $x \in \mathbb{R}_+^E$. $0 \leq x \leq 1$
- **Question**. Trouver X qui minimise $\min_{X \in \mathcal{C}} \sum_{e \in \delta(X)} x_e - 2$

et renvoie "oui" si x satisfait toutes les contraintes $\sum_{e \in \delta(X)} x_e \geq 2$ pour $X \in \mathcal{C}$. Si non, renvoie $X' \in \mathcal{C}$ tel que $\sum_{e \in \delta(X')} x_e < 2$.

Génération de ligne pour le TSP / Branch-and-cut

→ On fait un branch-and-bound. Pour résoudre la relaxation linéaire de (P) :
On choisit $\mathcal{C}' \subseteq \mathcal{C}$.

1. On résout

$$\begin{array}{lll} \min & \sum_{e \in E} c(e) x_e & \\ \text{s.c.} & 0 \leq x_e \leq 1 & \forall e \in E, \\ & \sum_{e \in \delta(v)} x_e = 2 & \forall v \in V, \\ & \sum_{e \in \delta(X)} x_e \geq 2 & \forall X \in \mathcal{C}'. \end{array}$$

Cela donne une solution $\bar{x}$.

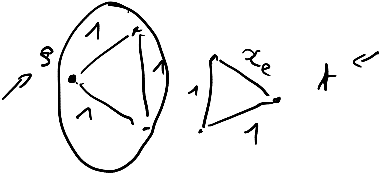
2. On applique $\mathcal{A}$ sur $\bar{x}$.
   - Si 'oui', alors on continue le branch-and-bound (*on a résolu la relaxation linéaire de* (P)).
   - Si non, on retourne en 1. en ajoutant le X' obtenu par $\mathcal{A}$ à $\mathcal{C}'$.

Séparation pour TSP

$$\min_{X \in \mathcal{C}} \sum_{e \in \delta(X)} x_e \qquad x_e \in [0,1]$$

Proposer un algorithme de séparation polynomial pour le TSP

$\forall\ s \in V,\ t \in V.$



Séparation pour TSP

Proposer un algorithme de séparation polynomial pour le TSP

L'algorithme $\mathcal{A}$ de séparation des contraintes pour le TSP :

Mettre $\bar{x}_e$ comme capacité de e, pour $e \in E$.

A s fixé : calculer pour tout t dans $V \setminus \{s\}$ une s-t coupe de capacité minimum.

Si l'une de ces coupes $\delta(X')$ a une capacité < 2, alors

$$X' \in \mathcal{C} \text{ et } \sum_{e \in \delta(X')} \bar{x}_e < 2.$$

Recherche locale

Proposer un voisinage pour le TSP

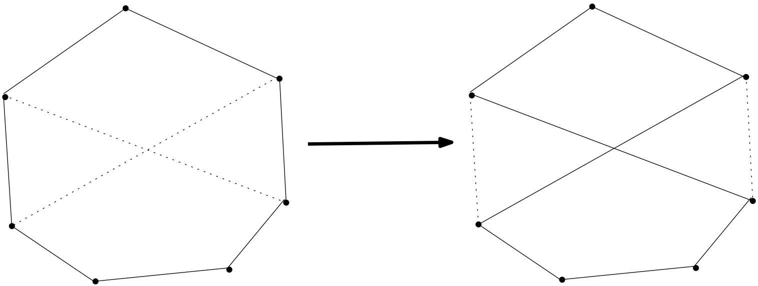
Recherche locale

Proposer un voisinage pour le TSP

Voisinage 2-OPT : le plus classique. Deux cycles hamiltoniens C, C' sont voisins s'ils différent d'exactement deux arêtes $|C \setminus C'| = |C' \setminus C| = 2$.

On peut alors intégrer ce voisinage dans n'importe quel schéma de métaheuristique du type "recherche locale" (méthode tabou, recuit simulé,...).

Illustration



Postier

Problème du postier

- **Instance**. Graphe $G = (V, E)$, fonction de coût sur les arêtes $c : E \to \mathbb{R}_+$.
- **Question**. Trouver une chaîne fermée C passant par toutes les arêtes telle que $\sum_{e \in C} c(e)$ minimum.

Postier

Problème du postier

- **Instance**. Graphe $G = (V, E)$, fonction de coût sur les arêtes $c : E \to \mathbb{R}_+$.
- **Question**. Trouver une chaîne fermée C passant par toutes les arêtes telle que $\sum_{e \in C} c(e)$ minimum.

Théorème. C'est un problème polynomial.

Exercice : problème du postier orienté

Problème du postier orienté

- **Instance**. Graphe orienté $G = (V, A)$, fonction de coût sur les arcs $c : A \to \mathbb{R}_+$.
- **Question**. Trouver un chemin fermé C passant par tous les arcs telle que $\sum_{a \in C} c(a)$ minimum.

Proposer un algorithme polynomial pour résoudre le problème.

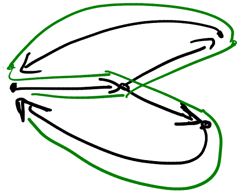
b - flot

$l = 1$

$u = +\infty$

$c = c$

$b = 0$



Exercice : problème du postier orienté

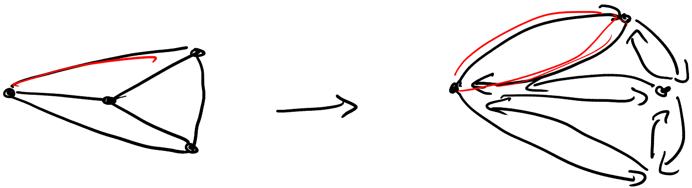
Problème du postier orienté

- **Instance**. Graphe orienté $G = (V, A)$, fonction de coût sur les arcs $c : A \to \mathbb{R}_+$.
- **Question**. Trouver un chemin fermé C passant par tous les arcs telle que $\sum_{a \in C} c(a)$ minimum.

Proposer un algorithme polynomial pour résoudre le problème.

On définit des capacité $\ell(a) = 1$ et $u(a) = +\infty$, et $b(v) = 0$. On cherche un b-flot de coût minimum (dans ce cas $b = 0$ c'est une circulation)

• Cas non orienté : pourquoi on ne peut pas doubler et orienter chaque arête ?



Cas non orienté

Rappel : graphes eulériens

Il y a un cas où c'est facile :

Théorème. Un graphe connexe admet une chaîne eulérienne si et seulement si il possède au plus deux sommets de degré impair.

Un graphe connexe admet un cycle eulérien si et seulement si il n'a pas de sommet de degré impair.

1 Un tel graphe est eulérien
Graphe eulérien : optimum = somme des coûts des arêtes.

Si le graphe n'est pas eulérien

Attention, on suppose $c_e \geq 0$.

Lemma

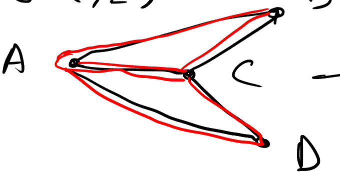
Il existe un parcours du postier optimal qui passe au plus deux fois par chaque arête.

• $G = (V, E)$

Soit une solution du pb de postier.

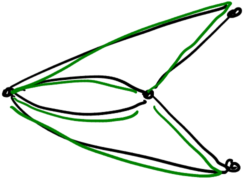
• $G' = (V, F)$

$G = (V, E)$

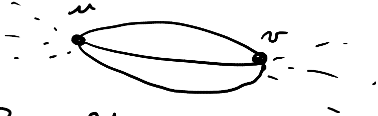


$\longrightarrow$

$G' = (V, F)$ est eulerien

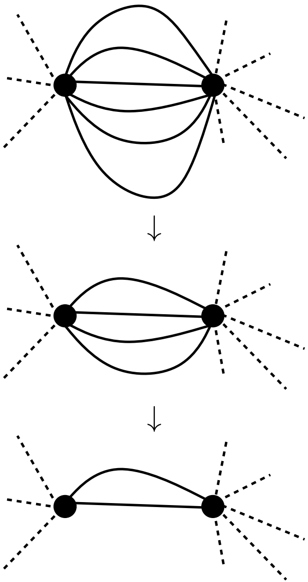


•

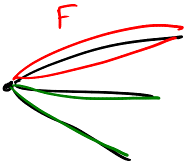


donc $deg(u)$, $deg(v)$ pairs

Si plus que 2 arêtes, on peut en enlever 2 et le graphe reste eulérien.



Reformulation



D'où la reformulation

Problème du postier

- **Instance**. Un graphe $G = (V, E)$ et une fonction de coût sur les arêtes $c : E \to \mathbb{R}_+$.
- **Question**. Trouver $F \subseteq E$ tel que $\deg_F(v) = \deg_E(v) \bmod 2$ pour tout $v \in V$, et tel que $\sum_{e \in F} c(e)$ soit minimal.

Comment reconstruit-on la tournée à partir de la solution ?

Reformulation

D'où la reformulation

Problème du postier

- **Instance**. Un graphe $G = (V, E)$ et une fonction de coût sur les arêtes $c : E \to \mathbb{R}_+$.
- **Question**. Trouver $F \subseteq E$ tel que $\deg_F(v) = \deg_E(v) \bmod 2$ pour tout $v \in V$, et tel que $\sum_{e \in F} c(e)$ soit minimal.

Comment reconstruit-on la tournée à partir de la solution ?

Dans le graphe où on a 2 copies des arêtes de F : il existe un circuit Eulérien.

De quoi est composé un graphe F optimal ? A quel type de problème peut-on donc réduire ce problème ?

Reformulation, suite

$F \subseteq E$ tel que $\deg_F(v) = \deg_E(v) \bmod 2$ pour tout $v \in V$ et tel que $\sum_{e \in F} c(e)$ soit minimal

Si les $c(e) \geq 0$, un tel F est constitué de chaînes appariant les sommets de degré impairs de G.

Trouver un tel F :

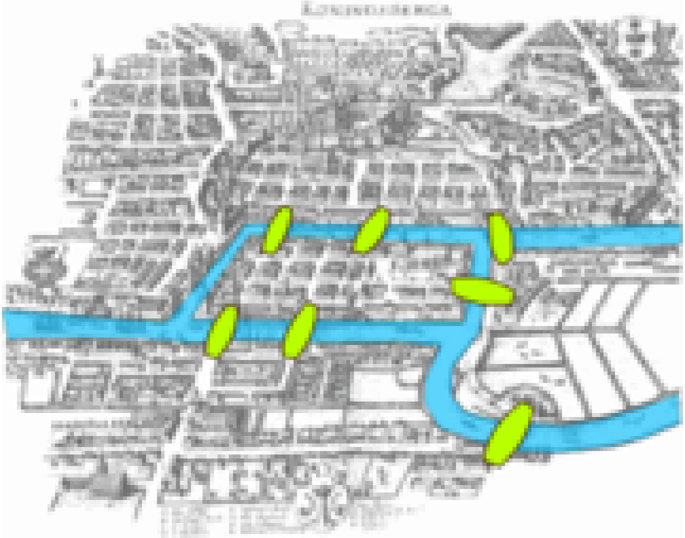
- soit J l'ensemble des sommets de degré impair de G,
- soit K le graphe complet sur J,
- on définit

  $c(jj') =$ le coût de la chaîne de plus petit coût entre j et j'

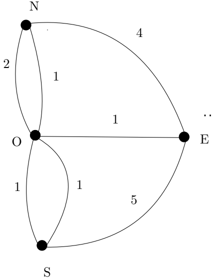
  sur chaque arête jj' de K,
- le couplage parfait de plus petit coût donne alors la solution.

Exemple : Retour sur les ponts de Königsberg

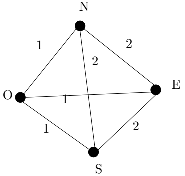
Euler, 1736.



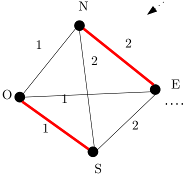
Exemple : Retour sur les ponts de Königsberg



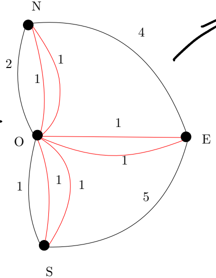
graphe complet des plus courtes chaînes



cycle eulérien



couplage parfait de poids minimum



Pour aller plus loin

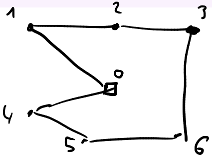
Problèmes de tournées à plusieurs véhicules.

Beaucoup plus difficile

Approches exactes à bases de PLNE : découpler pour obtenir un sous-problème par véhicule

- relaxation Lagrangienne
- génération de colonnes (pour traiter un nombre exponentiel de variables)

Exercice 1 : TSP with time windows



$[a_i, b_i]$

Un représentant de commerce doit visiter des clients situés en des villes différentes. Il a fixé un rendez-vous pour chacun de ses clients, i.e. pour tout client i, le représentant sait qu'il doit passer après l'instant a_i, mais avant l'instant b_i. Il souhaite optimiser sa tournée (et en passant vérifier que les contraintes ne sont pas contradictoires). On suppose qu'il sait le temps qu'il lui faut pour aller d'une ville à une autre d_{ij}. On part du dépôt d'indice 0, et on revient au dépôt à la fin de la tournée.

Proposer une formulation sous la forme d'un programme linéaire mixte (variables entières et continues), qui, contrairement au problème du voyageur de commerce usuel, ne contient pas un nombre exponentiel de contraintes.

- variables : • t_i temps de visite de i $t_i \in \mathbb{R}_+$
  - t_0 temps de retour en O
  - $\delta_{ij} \in \{0,1\}$ $\left| \begin{array}{l} = 1 \text{ si } i \text{ avant } j \text{ dans le tour} \\ = 0 \text{ sinon} \end{array} \right.$

- <u>Objectif</u> : $\min t_0$

$\left(\begin{array}{c} \text{or } \forall i < j \\ \delta_{ij} = 1 - \delta_{ji} \end{array} \right)$

- $a_i \leq t_i \leq b_i$, $\forall i \in [\![1, n]\!]$

  $t_0 \geqslant 0$

(• Si $i \longrightarrow j$, $t_j \geqslant t_i + d_{ij}$)

$\left| \begin{array}{l} M \geqslant 0 \\ \text{grand} \end{array} \right.$

- $-d_{ji} + M\delta_{ij} \geqslant t_j - t_i \geq d_{ij} - M(1 - \delta_{ij})$

  $\forall i < j$

- $\forall i \quad t_i \geq 0 + d_{0i}$ → départ du dépôt

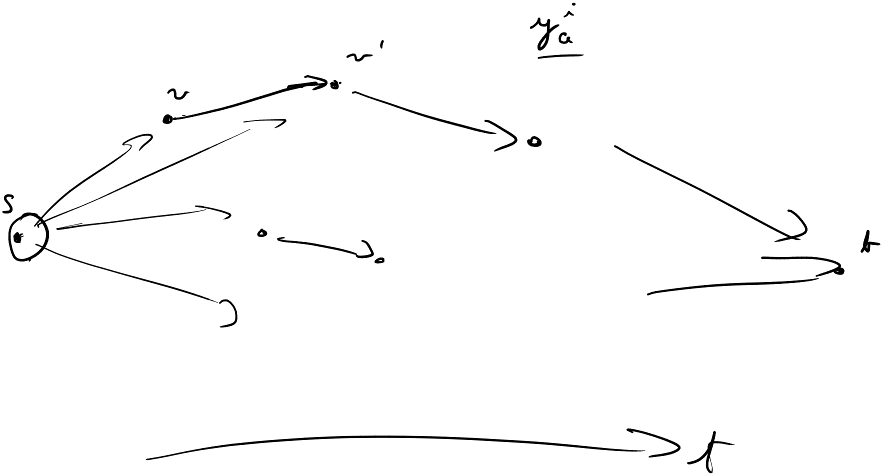
  $\forall i \quad t_0 \geq t_i + d_{i0}$ → retour au dépôt

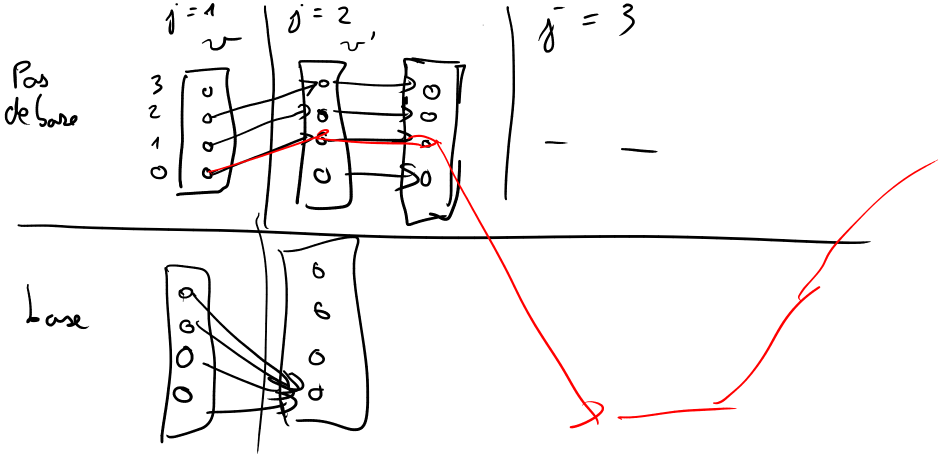
Exercice 2 : Aircraft routing

Un compagnie aérienne souhaite réaliser les tournées de ses avions de manière à opérer tous ses vols à coût minimum. Soit V l'ensemble des vols de la compagnie aérienne. Soit I l'ensemble des avions. On note c_{vi} le coût d'opérer v avec i. De manière à pouvoir faire des maintenances de sécurité, un avion doit passer une nuit à la base (un aéroport spécialement équipé) au moins tous les quatre jours.

1) Proposez une formulation PLNE pour modéliser ce problème (avec un nombre polynomial de variables).

2) Soit $\mathcal{P}$ l'ensemble des séquences de vols qui peuvent être faites par les avions. Proposez une autre formulation PLNE utilisant les variables $x_P^i \in \{0, 1\}$. Expliquer comment résoudre une telle formulation.


y_a^i
s
v
v'
t
t



$$\begin{cases} \min \sum_p c_p x_p \\ \text{sc} \quad \sum_{p \ni v} x_p = 1 \quad \forall v \end{cases}$$