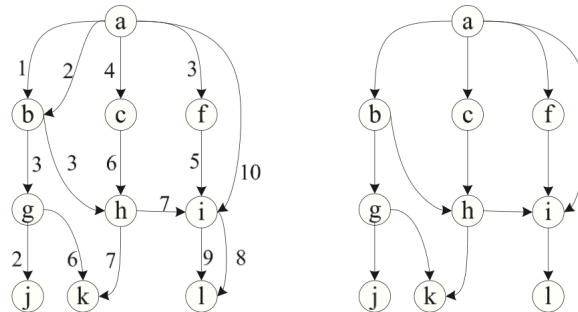


Projet

Nous allons considérer des multigraphes orientés pondérés par le temps (dans un multigraphe plusieurs arcs sont possibles entre deux sommets). Soit G un tel multigraphe. On va supposer que G est sans circuit. Supposons que G soit un réseau de transport aérien où chaque sommet représente un aéroport et le nombre sur chaque arc soit le jour (date) de départ d'un vol dont la ville de départ est l'extrémité initiale et la destination est l'extrémité finale de l'arc.

Exemple 1 Pour simplifier, dans cet exemple nous supposons que la durée de chaque vol est de 1 jour. La figure de gauche montre un tel réseau de transport aérien. Les 2 arcs de a à b indiquent qu'il y a un vol de a à b le jour 1 et un autre le jour 2, c'est-à-dire que les valeurs 1 et 2 sur les arcs représentent la date de départ du vol.



La figure de droite montre le graphe statique condensé G_S de G . Nous pouvons voir que certains chemins dans le graphe statique peuvent ne pas être un chemin significatif dans le multigraphe initial.

Par exemple, (a, b, g, j) est un chemin dans la figure de droite, mais (a, b, g, j) dans la figure de gauche est problématique car g n'a qu'un seul vol vers j le jour 2 mais nous ne pouvons pas atteindre g avant le jour 4 (en partant de b le jour 3 et en prenant 1 jour pour aller de b à g). Considérons maintenant un plus court chemin de a à l . Dans la figure de droite, le plus court chemin est (a, i, l) avec une distance de 2. Mais dans le multigraphe G de la figure de gauche, si nous prenons l'arc (a, i) , alors nous ne pouvons prendre aucun vol de i à l puisque le vol de a arrive à i le jour 11. Au lieu de cela, un chemin valide dans G respectant les contraintes de vols est (a, f, i, l) avec une distance de 3 : on partirait de a à f le jour 3, de f à i le jour 5, et de i à l le jour 8.

En raison de la présence de contraintes au niveau du temps, différents types de “plus courts” chemins existent et chacun a sa propre importance. Nous définissons quatre types de chemins dans un multigraphe pondéré G ; nous les appelons chemins minimaux, car ils donnent la valeur minimale pour différentes mesures: (1) chemin d'arrivée au plus tôt (c'est-à-dire un chemin qui

donne la date d'arrivée au plus tôt en partant d'une source x vers une cible y) ; (2) chemin de départ au plus tard (c'est-à-dire un chemin qui donne la date de départ au plus tard en partant de x afin d'atteindre y avant un temps donné) ; (3) le chemin le plus rapide (c.-à-d. le chemin par lequel on va de x à y avec le temps écoulé le plus court) ; et (4) le chemin le plus court (c.-à-d. le chemin le plus court de x à y en termes de temps de traversée total nécessaire).

Plus formellement, soit $G = (V, E)$ un multigraphe orienté avec des contraintes de temps, où V est l'ensemble des sommets de G et E est l'ensemble des arcs de G . Un arc $e \in E$ est un quadruple (u, v, t, λ) , où $u, v \in V$, t est le temps de départ, λ est le temps de traversée pour aller de u à v en commençant au temps t , et $t + \lambda$ est le temps de fin. Nous désignons le temps de départ de e par $t(e)$ et le temps de traversée de e par $\lambda(e) > 0$.

Un chemin P dans un multigraphe pondéré par le temps G est une séquence de sommets $P = (v_1, v_2, \dots, v_k, v_{k+1})$, où $(v_i, v_{i+1}, t_i, \lambda_i) \in E$ est le i -ième arc sur P pour $1 \leq i \leq k$, et $(t_i + \lambda_i) \leq t_{i+1}$ pour $1 \leq i < k$. Notez que pour la dernière arête $(v_k, v_{k+1}, t_k, \lambda_k)$ sur P , nous ne mettons pas de contrainte sur $(t_k + \lambda_k)$. On dira que $(t_k + \lambda_k)$ est la date de fin de P , notée $\text{fin}(P)$. Nous définissons également la date de début de P comme $\text{début}(P) = t_1$. Nous définissons la durée de P comme $\text{durée}(P) = \text{fin}(P) - \text{début}(P)$. Enfin, la distance est définie comme : $\text{dist}(P) = \sum_{i=1}^k \lambda_i$.

Dans le multigraphe G de la figure de gauche un chemin est $P = ((a, f, 3, 1), (f, i, 5, 1), (i, l, 8, 1))$. Nous avons $\text{début}(P) = 3$, $\text{fin}(P) = 8 + 1 = 9$, $\text{durée}(P) = 9 - 3 = 6$ et $\text{dist}(P) = 1 + 1 + 1 = 3$.

Étant donné un multigraphe pondéré par le temps G , un sommet source x et un sommet cible y dans G , et un intervalle de temps $[t_\alpha, t_\omega]$, soit

$$\mathcal{P}(x, y, [t_\alpha, t_\omega]) = \{P : P \text{ est un chemin de } x \text{ à } y \text{ tel que } \text{début}(P) \geq t_\alpha, \text{fin}(P) \leq t_\omega\}.$$

Nous définissons formellement les quatre types de chemins de x à y dans $[t_\alpha, t_\omega]$:

- **Type I : Chemin d'arrivée au plus tôt** : $P \in \mathcal{P}(x, y, [t_\alpha, t_\omega])$ est un chemin d'arrivée au plus tôt si $\text{fin}(P) = \min\{\text{fin}(P') : P' \in \mathcal{P}(x, y, [t_\alpha, t_\omega])\}$.
- **Type II : Chemin de départ au plus tard** : $P \in \mathcal{P}(x, y, [t_\alpha, t_\omega])$ est un chemin de départ au plus tard si $\text{début}(P) = \max\{\text{début}(P') : P' \in \mathcal{P}(x, y, [t_\alpha, t_\omega])\}$.
- **Type III : Chemin le plus rapide** : $P \in \mathcal{P}(x, y, [t_\alpha, t_\omega])$ est un chemin le plus rapide si $\text{durée}(P) = \min\{\text{durée}(P') : P' \in \mathcal{P}(x, y, [t_\alpha, t_\omega])\}$.
- **Type IV : Plus court chemin** : $P \in \mathcal{P}(x, y, [t_\alpha, t_\omega])$ est un plus court chemin si

$$\text{dist}(P) = \min\{\text{dist}(P') : P' \in \mathcal{P}(x, y, [t_\alpha, t_\omega])\}.$$

Question 1. En utilisant l'instance de la figure gauche de l'exemple 1 ou une autre instance, montrer que les assertions suivantes sont vraies.

Dans un multigraphe orienté sans circuit avec des contraintes de temps :

Assertion 1 : Un sous-chemin préfixe d'un chemin d'arrivée au plus tôt peut ne pas être un chemin d'arrivée au plus tôt.

Assertion 2 : Un sous-chemin postfixe d'un chemin de départ au plus tard peut ne pas être un chemin de départ au plus tard.

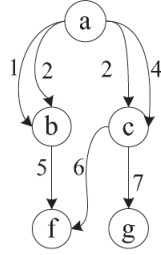
Assertion 3 : Un sous-chemin d'un chemin le plus rapide peut ne pas être un chemin le plus rapide.

Assertion 4 : Un sous-chemin d'un plus court chemin peut ne pas être un plus court chemin.

Nous proposons maintenant une technique de transformation du multigraphe graphe pondéré par le temps à un graphe classique. Cette transformation devrait nous inspirer afin de concevoir des algorithmes pour calculer les quatre types de chemins minimaux précités. Nous présentons d'abord comment transformer un multigraphe orienté sans circuit pondéré par les contraintes de temps $G = (V, E)$ en un graphe $\tilde{G} = (\tilde{V}, \tilde{E})$ qui prend en compte le temps de manière explicite.

Ainsi pour chaque sommet $v \in V$, on crée dans \tilde{V} un nombre de sommets égal au nombre de dates différentes auxquelles on peut arriver au sommet v par un arc entrant à v . On appelle cet ensemble de sommets $\tilde{V}_{in}(v)$. Chaque sommet introduit, que l'on appellera aussi copie de v , sera étiqueté par (v, t) , t étant la date à laquelle on arrive à v par un des arcs entrants du graphe initial G . De la même manière, on crée dans \tilde{V} un nombre de sommets égal au nombre de dates différentes auxquelles on peut partir du sommet v . Chaque sommet introduit (v, t) correspond aux différentes dates auxquelles on peut quitter le sommet v en utilisant un arc sortant de v . On appelle cet ensemble des sommets $\tilde{V}_{out}(v)$.

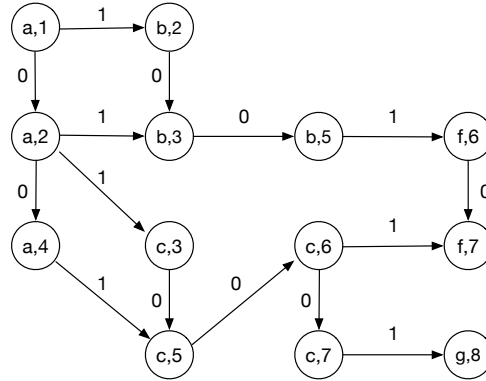
Exemple 2 Considérons le multi graphe orienté pondéré par le temps G de la figure suivante. Comme dans l'exemple 1, nous supposons que la durée de chaque vol est de 1 jour.



Le tableau suivant donne les ensembles $\tilde{V}_{in}(v)$ et $\tilde{V}_{out}(v)$ pour chaque sommet :

	$\tilde{V}_{in}(v)$	$\tilde{V}_{out}(v)$
a	—	$\{(a, 1), (a, 2), (a, 4)\}$
b	$\{(b, 2), (b, 3)\}$	$\{(b, 5)\}$
c	$\{(c, 3), (c, 5)\}$	$\{(c, 6), (c, 7)\}$
f	$\{(f, 6), (f, 7)\}$	—
g	$\{(g, 8)\}$	—

Pour tout sommet $v \in V$ du multigraphe initial, soit $\tilde{V}(v) = \tilde{V}_{in}(v) \cup \tilde{V}_{out}(v)$, l'ensemble des sommets insérés dans \tilde{G} lors de sa construction. Dans le nouveau graphe \tilde{G} , l'ensemble des arcs \tilde{E} est défini de la manière suivante : pour tout sommet $v \in V$, soit $\tilde{V}(v) = \{(v, t_1), (v, t_2), \dots, (v, t_k)\}$, avec $t_1 < t_2 < \dots < t_k$ et $k \geq 1$, les sommets insérés dans \tilde{V} . On insère les arcs $(v, t_i) \rightarrow (v, t_{i+1})$, pour tout $1 \leq i < k$ avec un poids égal à 0. Maintenant pour chaque arc $(u, v, t, \lambda) \in E$ du multigraphe initial G , on insère dans \tilde{E} un arc à partir du sommet (u, t) vers le sommet $(v, t + \lambda)$ avec un poids égal à λ . Pour le graphe de l'exemple 2, nous obtenons le graphe \tilde{G} suivant :



Question 2. En utilisant comme base cette transformation de G à \tilde{G} , montrer comment on peut calculer de manière efficace les 4 types de chemins minimaux définis précédemment. Il sera peut-être nécessaire d'adapter \tilde{G} de manière appropriée.

Question 3. Calculer la complexité de différents algorithmes proposés.

Question 4. Ecrire un programme qui prend en entrée un multigraphe sans circuit G et retourne les 4 types de chemins minimaux correspondants. Votre programme doit utiliser les algorithmes proposés dans la question 2. On devra donner la possibilité de saisir le multigraphe sur le terminal et en lisant un fichier. Le format de fichier d'entrée doit respecter le format suivant :

```

n % nombre de sommets
m % nombre d'arcs
sommet 1
sommet 2
...
sommet n
arc 1 % sous la forme (u,v,t,lambda)
arc 2
...
arc m

```

Question 5. Proposer une modélisation du problème de plus court chemin par programmation linéaire et implanter une méthode de recherche de chemin de Type IV en faisant appel à GUROBI.

Question 6. Effectuer des tests pour mesurer le temps d'exécution de votre algorithme par rapport à la taille de l'entrée (nombre de sommets, nombre d'arcs, étiquettes sur les sommets).

Question 7. Comparer les algorithmes implantés pour le calcul de chemins de type IV dans les questions 4 et 5.

Question subsidiaire : Proposer et implanter des algorithmes alternatifs pour calculer les chemins minimaux sans passer par la transformation de graphe. Expliquer pourquoi votre algorithme est correct, calculer sa complexité, comparer ses performances par rapport aux algorithmes qui utilisent la transformation de graphe.

Fonctionnement et échéances et livrables

Le projet est à réaliser en binôme, le binôme devant être composé de deux personnes inscrites dans le même groupe en MOGPL. Les binômes devront remettre un document pdf où seront rédigées les réponses aux questions du projet. On mettra en annexe de ce document pdf une description des modèles et tests réalisés ainsi qu'une archive zip concernant les programmes réalisés. Ces documents seront remis au chargé de TD de votre groupe (un par binôme) via une soumission Moodle au plus tard le 6 décembre 2021. Les soutenances de projet auront lieu lors des créneaux habituels de TD la semaine du 13 décembre 2021.