Master TET, ENPC 2025-26
Recherche Opérationnelle (TEOST/R2O)

TD1 : Modélisation en optimisation linéaire

Exercice 1. *Distribution d'énergie.*

Une société de production d'énergie possède deux centrales électriques C_1 et C_2 qui alimentent trois villes V_1, V_2 et V_3. Chaque centrale a une capacité de production limitée, et chaque ville a une demande spécifique en énergie. Les coûts de transport de l'énergie entre chaque centrale et chaque ville sont connus (cf. tableau ci-dessous). L'objectif est de déterminer la distribution d'énergie optimale pour minimiser le coût total de transport tout en respectant les capacités des centrales et les demandes des villes.

Modéliser ce problème sous forme de programmation linéaire (PL).

Données.

- Capacités de production des centrales :
  - C_1 : 600 MWh
  - C_2 : 400 MWh
- Demandes des villes :
  - V_1 : 300 MWh
  - V_2 : 200 MWh
  - V_3 : 400 MWh
- Coûts de transport (€/MWh) entre centrales et villes :

	V_1	V_2	V_3
C_1	4	6	9
C_2	5	4	7

Exercice 2. *Un problème de production.*

Une usine de production d'hydrogène dispose de deux procédés différents P_1 et P_2 de production nécessitant 3 ressources : équipement, énergie et matières premières, disponibles en quantités limitées (dernière colonne du tableau ci-dessous). Le tableau suivant indique les besoins en ressources pour produire 1kg d'H_2.

	P_1	P_2	disponibilité
équipement	3	8	72
énergie	4	5	50
matière première	2	1	20

Ainsi, pour produire 1kg d'H_2, le procédé P_1 a besoin de 3 unités d'équipements, 4 unités d'énergie et 2 de matières premières.

1. Modéliser ce problème sous forme de PL pour déterminer les quantités (en kg) de produits P_1 et P_2 à fabriquer afin de maximiser la production totale d'H_2.
2. Que faut-il changer dans la modélisation précédente si l'hydrogène est produite dans des conteneurs de 100kg (et non plus au poids) ?

Exercice 3. *Gestion de projets*

Les n étudiants d'une formation doivent réaliser des projets individuels. Chaque étudiant doit réalisé un et un seul projet. De même, chaque projet doit être attribué à un étudiant et un seul. Pour décider de la répartition des projets, les étudiants choisissent 3 projets et les classent par ordre strictement croissant de préférence. Les projets retenus par un étudiant se voient ainsi attribués une note de 1 à 3 et on considère que les projets non-retenus ont la note 0. On cherche à maximiser la satisfaction générale définie comme étant la somme (ou la moyenne) des notes données par les étudiants aux projets auxquels ils sont affectés.

1. Modéliser ce problème sous forme de programmation linéaire, afin de déterminer une répartition optimale des projets en maximisant la satisfaction générale. Vous introduirez une matrice E des notes projets/étudiants dont les coefficients sont les notes données à chaque projet par chaque étudiant.
2. Pour $n = 4$, donner un exemple de matrice E des notes projets/étudiants. Pour $n = 4$, écrire le problème de programmation linéaire obtenu précédemment, sous la forme

$$\begin{aligned} &\max_{\mathbf{x}} \left[F(\mathbf{x}) = \mathbf{e}^\top \mathbf{x} \right] \\ &\begin{cases} A\mathbf{x} = \mathbf{b} \\ \mathbf{x} \in \{0,1\}^{n^2} \end{cases} \end{aligned} \tag{1}$$

   en précisant la variable $\mathbf{x}$ et en explicitant les vecteurs $\mathbf{e}$ et $\mathbf{b}$ ainsi que la matrice A.
3. La modélisation précédente souffre d'un manque de considération individuelle... Dans la répartition optimale précédente, des étudiants peuvent très bien se voir attribuer des projets qu'il n'avaient pas classés initialement (i.e. avec la note 0). On décide alors que chaque étudiant doit être affecté à un projet parmi les 3 projets classés initialement.
   (a) Modéliser cette nouvelle contrainte.
   (b) Pour $n = 4$, écrivez cette contrainte sous la forme

$$B\mathbf{x} \geq \mathbf{d} \tag{2}$$

   en explicitant la matrice B et le vecteur $\mathbf{d}$.
   (c) Donner un exemple de matrice des notes projet/étudiant pour laquelle il n'y a pas de solution réalisable au problème (1),(2).

Exercice 4. *Plans de conservation de la biodiversité et linéarisation d'un terme quadratique*

On considère K espèces vivantes susceptibles de disparaître (si aucune disposition n'est prise) et n sites intéressants à éventuellement préserver (car chacun contenant tout ou partie de ces espèces) mais la préservation de chaque site i a un coût évalué à c_i. On estime aussi à $r_{ki} \geq 0$ la population de l'espèce k du site i. On cherche à préserver une population totale au moins égale à t_k pour chaque espèce k.

1. Écrire le PL qui consiste à réaliser cet objectif à moindre coût en sélectionnant les sites à préserver (on ne peut pas préserver tous les sites...). Utiliser les n variables binaires x_i ($x_i = 1$ pour prendre le site i dans le plan de conservation et $x_i = 0$ sinon).
2. À la fonction linéaire précédente on peut ajouter un terme quadratique du type :

$$\tau \sum_{i=1}^{n} \sum_{j=i+1}^{n} v_{ij} x_i x_j$$

   où le scalaire $\tau > 0$ permet de régler l'action plus ou moins forte de ce terme par rapport au terme linéaire. Ce terme peut avoir différentes fonctions. Par exemple avec $v_{ij} := d_{ij}$ où d_{ij} est la distance entre les sites i et j, on va avoir tendance à sélectionner des sites proches mais avec $v_{ij} := 1/d_{ij}$ on chercherait à avoir une certaine dispersion des sites choisis. Il est plus difficile de résoudre ce type de problèmes mais en augmentant le nombre de variables et de contraintes, on peut de nouveau retrouver un PL(NE) en posant

$$z_{ij} := x_i x_j$$

   Questions :
   (a) Combien utilise-t-on de variables z_{ij} ?
   (b) Trouver les contraintes d'inégalités (linéaires) (3 inégalités par paire $\{i, j\}$) qui permettent de relier les valeurs de x_i, x_j et z_{ij}.

(c) En déduire le PL obtenu avec cette astuce.

Remarque. Le terme quadratique peut aussi être de la forme

$$-\tau \sum_{\{i,j\}\in\mathcal{I}} v_{ij}x_ix_j$$

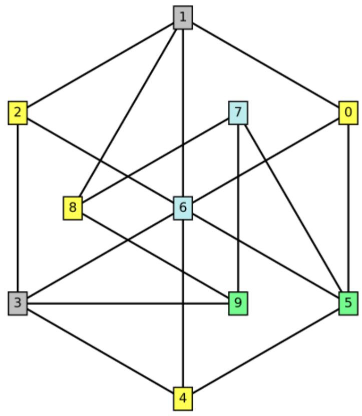
où $\mathcal{I}$ est l'ensemble des paires de sites $\{i,j\}$ qui sont intéressants à sélectionner ensemble (par exemple, une rivière traverse à la fois le site i et le site j, le scalaire $v_{ij} > 0$ mesurant l'intérêt de cette agrégation). Dans ce cas, le nombre de variables additionnelles est égal à $card(\mathcal{I})$ et donc généralement bien moindre que précédemment (de même que le nombre de contraintes additionnelles).

Exercice 5. *Coloration de graphe*

On veut attribuer une couleur à n villes constituées en réseau représenté par un graphe. Dans ce réseau, il existe des liaisons directes (les arêtes) entre villes (les sommets), mais toutes les villes ne sont pas reliées directement entre elles. On dispose de m couleurs différentes et le coloriage doit se faire en respectant les contraintes suivantes.

(a) Toutes les villes sont coloriées et il y a une seule couleur par ville.

(b) On ne peut pas colorier de la même couleur 2 villes qui sont reliées directement entre elles.



Le réseau est défini par une matrice de taille $n \times n$ dont les coefficients e_{ij} sont donnés par

$$e_{ij} = \begin{cases} 1 & \text{si les villes } i \text{ et } j \text{ sont reliées directement entre elles,} \\ 0 & \text{sinon.} \end{cases}$$

On veut minimiser le nombre de couleurs utilisées en respectant les contraintes.

1. Modéliser ce problème sous forme d'un programme linéaire en variables binaires. Vous introduirez des variables binaires x_{ik} permettant d'établir le coloriage des villes, ainsi que des variables y_k indiquant si la couleur k est utilisée.
(*indication* : pour la contrainte (b), vous pourrez considérer des termes de la forme $x_{ik}+x_{jk}$)
2. Ecrire le problème obtenu précédemment sous la forme suivante :

$$\min_{\mathbf{z}} \left[F(\mathbf{z}) = \mathbf{c}^\top \mathbf{z}\right]$$
$$\begin{cases} A\mathbf{z} = \mathbf{b} \\ B\mathbf{z} \leq \mathbf{d} \\ \mathbf{z} \in \{0,1\}^N, \end{cases}$$

avec $\mathbf{z} = (x_{11}, x_{21}, \cdots, x_{n1}|, \cdots, |x_{1m}, x_{2m}, \cdots, x_{nm}, |y_1, \cdots, y_m)^\top$ et $N = nm + m$.
Donner les matrices A, B et les vecteurs $\mathbf{b}$, $\mathbf{c}$ et $\mathbf{d}$ en précisant bien les dimensions.

Exercice 6. *Mariages stables et PL*

Dans le problème des mariages stables avec n couples homme/femme, un mariage est dit *instable* s'il y a au moins un homme et une femme qui préféreraient se mettre en couple plutôt que de rester avec leurs partenaires actuels. On cherche à former n couples stables (aucun couple instable).

On dispose de deux ensembles $H = \{h_1, \dots h_n\}$ (les hommes) et $F = \{f_1, \dots, f_n\}$ (les femmes). Chaque individu a un ordre de préférence *strict* (noté $\succ$) sur les membres de l'autre ensemble. On définit alors :

$P_h(f)$ l'ensemble des hommes que la femme $f \in F$ préfère à $h \in H$.
$P_f(h)$ l'ensemble des femmes que l'homme $h \in H$ préfère à $f \in F$.

On cherche à former les n couples (h, f) tels que :

1. Chaque homme est marié à une unique femme et chaque femme à un unique homme,
2. Il n'existe pas de paire bloquante (h_i, f_j) telle que :
   — h_i préfère f_j à sa partenaire actuelle
   — f_j préfère h_i à son partenaire actuel.

Pour modéliser ce problème par programmation linéaire, on introduit les variables binaires

$$x_{ij} = \begin{cases} 1 & \text{si l'homme } h_i \text{ est en couple avec la femme } f_j \\ 0 & \text{sinon} \end{cases}$$

QUESTIONS.

1. *Un exemple d'ensemble de préférences.* On considère l'exemple suivant avec $n = 3$ et les préférences indiquées pour chaque individu h/f :

$$\begin{array}{c} h_1 : f_1 \succ f_2 \succ f_3 \\ h_2 : f_2 \succ f_3 \succ f_1 \\ h_3 : f_3 \succ f_1 \succ f_2 \\ f_1 : h_2 \succ h_1 \succ h_3 \\ f_2 : h_3 \succ h_2 \succ h_1 \\ f_3 : h_1 \succ h_3 \succ h_2 \end{array}$$

   Construire les ensembles $P_h(f)$ et $P_f(h)$ pour $h \in H$, $f \in F$ associés à cet exemple.

2. Modéliser les contraintes 1. et montrer que les inégalités

$$\sum_{j' \in P_{f_j}(h_i)} x_{ij'} + \sum_{i' \in P_{h_i}(f_j)} x_{i'j} + x_{ij} \quad \geq 1 \quad \forall (i, j).$$

   modélise bien les contraintes 2. de stabilité.

3. Dans la formulation standard du problème des mariages stables, il n'y a rien à optimiser. On peut envisager d'introduire un objectif d'optimisation. Par exemple, on cherche à maximiser le nombre d'hommes qui obtiennent leur premier choix. Modéliser cet objectif en considérant la femme $p_i \in F$ qui est le premier choix de l'homme h_i.

Exercice 7. *Optimisation d'entrepôts*

Une entreprise dispose d'un produit stocké dans ses n entrepôts pour approvisionner ses m services selon leurs demandes. Un entrepôt est soit ouvert soit fermé. Quand un entrepôt est ouvert il y a un coût fixe de fonctionnement et le transport du produit entre un entrepôt et un service comporte un coût unitaire.

Les coûts sont :

(a) f_i : coût fixe de fonctionnement de l'entrepôt i s'il est ouvert.

(b) c_{ij} : coût unitaire de transport du produit de l'entrepôt i au service j.

Les contraintes sont :

(1) Chaque service j demande une quantité d_j de produit. L'approvisionnement peut se faire à partir d'un ou de plusieurs entrepôts mais la demande de chaque service doit toujours être *exactement* pourvue.

(2) Le produit peut être acheminé d'un entrepôt à un service uniquement si l'entrepôt est ouvert. Chaque entrepôt peut fournir au plus une quantité M de produit.

L'entreprise cherche à déterminer les entrepôts qu'elle doit ouvrir et les quantités de produit (valeurs réelles) à transporter des entrepôts vers les services afin de minimiser le **coût total** correspondant aux coûts de fonctionnement et de transport.

1. Modéliser ce problème sous forme d'un programme linéaire en considérant les variables binaires y_i indiquant si l'entrepôt i est ouvert ou fermé ainsi que les quantités x_{ij} de produit à transporter de l'entrepôt i au service j.
2. Ecrire le problème précédent sous la forme matricielle :

$$\min_{(\mathbf{x},\mathbf{y})} \left[F(\mathbf{x},\mathbf{y}) = \mathbf{c}^\top \mathbf{x} + \mathbf{e}^\top \mathbf{y} \right]$$
$$\begin{cases} A\mathbf{x} = \mathbf{d} \\ B\mathbf{x} - C\mathbf{y} \leq \mathbf{0} \\ \mathbf{x} \geq \mathbf{0},\ \mathbf{y} \in \{0,1\}^n , \end{cases}$$

avec $\mathbf{x} = (x_{11}, x_{12}, \cdots, x_{1m}|, \cdots, |x_{n1}, x_{n2}, \cdots, x_{nm})^\top$ et $\mathbf{y} = (y_1, \cdots, y_n)^\top$. Donner les matrices A, B, C et les vecteurs $\mathbf{c}$, $\mathbf{d}$, $\mathbf{e}$ en précisant bien les dimensions.

Exercice 8. *Modélisation du problème du voyageur de commerce par programmation linéaire (Miller-Tucker-Zemlin, 1960).*

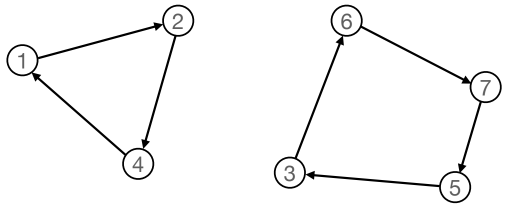
Le problème du voyageur de commerce (TSP Travelling Salesman Problem) s'énonce de la façon suivante. Etant données n villes, trouver le plus court *trajet* passant par toutes les villes une et une seule fois, en revenant à la ville de départ à la fin du trajet. Le but de cet exercice est de modéliser ce problème sous forme d'un programme linéaire. On note $d_{ij} \geq 0$ la distance séparant la ville i de la ville j (a priori $d_{ij} \neq d_{ji}$ pour $i \neq j$ et les valeurs d_{ii} soit valent $+\infty$, soit elles ne seront pas prises en compte dans tout ce qui suit). On introduit les variables d'affectation x_{ij} ($1 \leq i, j \leq n$) telles que

$$x_{ij} = \begin{cases} 1 & \text{si on va de la ville } i \text{ à la ville } j \\ 0 & \text{sinon.} \end{cases}$$

On suppose que le ville de départ (et d'arrivée) est la ville n$^{\text{o}}$1.

Préliminaires.

1. A partir des variables x_{ij}, définir l'objectif et les contraintes auxquelles sont soumises ces variables.
2. On peut représenter le TSP à l'aide d'un graphe où les sommets correspondent aux villes et les arêtes aux déplacements entre les villes dans le trajet. Dans le TSP, un trajet ne peut être constitué de *sous-circuits* (boucles) non reliés entre eux. Par exemple, le graphe ci-dessous présente 2 sous-circuits.



Pour cet exemple, écrire la matrice $X = (x_{ij})_{1 \leq i,j \leq 7}$ et montrer que les contraintes sur ces variables sont respectées. Cette solution réalisable ne respecte pourtant pas la définition d'un trajet (partant de 1, on ne revient pas en 1 après avoir visité toutes les villes).

Modélisation MTZ. Pour éviter de possibles *sous-circuits*, on introduit les variables supplémentaires $u_i \in \mathbb{N}$ pour $1 \leq i \leq n$, représentant la position de la ville i dans la tournée du voyageur. Comme on a supposé que la ville 1 est la ville de départ, on a donc $u_1 = 1$ et par ailleurs $u_i \in \{2, \cdots, n\}$ pour $i \in \{2, \cdots, n\}$.

3. Les variables x_{ij} et u_i sont liées par le fait que si $x_{ij} = 1$ alors nécessairement $u_i < u_j$. Montrer que cette contrainte peut se traduire algébriquement par

$$u_i - u_j + 1 \leq \alpha(1 - x_{ij}), \quad 2 \leq i \neq j \leq n \tag{1}$$

où α est un réel à déterminer.

4. Modéliser le problème du voyageur de commerce sous forme d'un programme linéaire pour les variables x_{ij} $(1 \leq i \neq j \leq n)$ et u_i $(2 \leq i \leq n)$.

5. Dans l'exemple précédent présentant deux sous-circuits, montrer que les contraintes (1) ne peuvent pas être toutes vérifiées en faisant la somme de ces contraintes pour le sous-circuit $(3, 6, 7, 5, 3)$.

6. De façon générale, on peut montrer que s'il existe un sous-circuit alors nécessairement celui-ci doit passer par la ville de départ n°1. Ceci implique qu'il ne peut exister qu'un seul circuit partant de la ville 1, passant par toutes les villes et revenant à la ville 1 de départ (un *trajet*). Pour établir cette propriété, supposer par l'absurde qu'il existe un sous-circuit avec k villes $(i_1, i_2, \cdots, i_k, i_1)$ qui ne passe pas par la ville 1 de départ (i.e $i_j \neq 1$ pour $j = 1, \cdots, k$). Montrer alors une contradiction avec les contraintes (1) concernées.