Cahier des Projets 2016-17

Vous trouverez dans ce document, en plus de la liste des sujets proposés, des informations et des conseils concernant le contenu des projets ainsi que l'organisation générale. *Merci de tout lire.*

Introduction

L'objectif principal de ces projets est de vous proposer une opportunité pour mettre en pratique l'ensemble des notions théoriques abordées en cours et les compétences pratiques acquises en TPs. Parmi les objectifs subsidiaires, mentionnons celui de vous initier à la résolution de *problèmes d'optimisation à variables entières et continues* (nous dirons, pour faire simple, problèmes d'optimisation à variables mixtes) [1].

Il existe diverses techniques dédiées à la résolution des problèmes d'optimisation à variables mixtes. Nous utiliserons la technique de la *relaxation lagrangienne*. La raison principale motivant le choix de cette technique est pédagogique. En effet, pour être appliquée les connaissances abordées en cours et en TPs sont suffisantes et aucune autre notion n'est requise. Nous aurons besoin, principalement, des notions d'optimisation linéaire et de dualité vues en cours.

Pour fixer les idées, un problème d'optimisation en variables mixtes se présente comme suit :

$$\begin{array}{ll} \min & f(x,y) \\ s.c. & \\ & (x,y) \in P, \\ & (x,y) \in \mathbb{Z}^n \times \mathbb{R}^m, \end{array} \tag{1}$$

où P est un polyèdre de $\mathbb{R}^{n+m}$. Les contraintes

$$x \in \mathbb{Z}^n,$$

sont dites *contraintes d'intégrité*. Ce sont des contraintes non linéaire.

La *relaxation continue* du problème (1) est le problème d'optimisation suivant :

$$\begin{array}{ll} \min & f(x,y) \\ s.c. & \\ & (x,y) \in P, \\ & (x,y) \in \mathbb{R}^n \times \mathbb{R}^m. \end{array}$$

Ainsi, la relaxation continue d'un problème d'optimisation en variables mixtes consiste à simplement ignorer ses contraintes d'intégrité. Le qualificatif *relaxation* vient du fait que l'ensemble des solutions entières du problème (1) est inclus dans l'ensemble des solutions réalisables de sa relaxation continue. D'où, la valeur d'une solution optimale de la relaxation continue du problème (1) constitue un minorant pour la valeur d'une solution optimale de (1). La différence entre la valeur d'une solution optimale du problème (1) et la valeur de sa relaxation continue est dite *saut d'intégrité*. En général, le saut d'intégrité pour le problème (1) est non nul [2].

Relaxation lagrangienne

Nous rappelons dans cette section quelques éléments du cours sur la notion de dualité lagrangienne.

1. Bien sûr, ces projets seront aussi une nouvelle occasion pour vous exercer au travail de groupe que nous espérons s'effectuer dans des conditions amicales.
2. Lors du premier TP vous avez traité un problème en variables entières pour lequel le saut d'intégrité est toujours nul. Ces problèmes sont ceux qui peuplent la classe $\mathcal{P}$!

Soit le problème d'optimisation générique suivant :

$$\begin{aligned} &\min f(x) \\ &s.c. \\ &\quad x \in \mathcal{X} \cap \mathcal{F}, \end{aligned} \tag{$\mathcal{G}$}$$

où $\mathcal{X}$ et $\mathcal{F}$ sont deux ensembles définis explicitement à l'aide de contraintes d'inégalité de type $\leq$.

Considérons la relaxation lagrangienne du problème $(\mathcal{G})$:

$$\theta(\pi) = \min \{L(x, \pi) : x \in \mathcal{F}, \pi \geq 0\},$$

où $(x, \pi) \to L(x, \pi)$ est la fonction de Lagrange associée au problème $(\mathcal{G})$. Le vecteur π correspond aux multiplicateurs associés aux contraintes définissant l'ensemble $\mathcal{X}$. Le dual lagrangien est le problème d'optimisation suivant :

$$\max \{\theta(\pi) : \pi \geq 0\}.$$

L'algorithme 1 ci-dessous est une implémentation de la méthode de sous-gradient pour l'optimisation du dual lagrangien du problème primal $(\mathcal{G})$.

Algorithm 1 A subgradient method to optimize the Lagrangian dual problem

Require:
- ε control parameter
- $\rho \in]0, 2[$ control parameter
- π^0 initial nonnegative vector (usually the nul vector)
- `iterLimit` number of maximum iterations
- `DualNoChangTOL` accepted number of iterations without change in the value of the dual function

Ensure: A local optimum of θ and an approximate primal solution $\hat{x}$

Set $k \leftarrow 1$ and $t \leftarrow 1$
Set $\beta^k \leftarrow -\infty$ and $\pi^k \leftarrow \pi^0$
repeat
  Solve the Lagrangian relaxation : $\theta\left(\pi^k\right) \leftarrow \min_{x \in \mathcal{F}} L\left(x, \pi^k\right)$
  Set $x^k \leftarrow \arg\min \left\{L\left(x, \pi^k\right) : x \in \mathcal{F}\right\}$
  for all $j \in J$ **do**
    $\gamma_j^k \leftarrow g_j\left(x^k\right)$
  end for
  if $\theta\left(\pi^k\right) > \beta^k$ **then**
    Set $\beta^{k+1} \leftarrow \theta\left(\pi^k\right)$
  else
    if $t \leq$ `DualNoChangTOL` **then**
      Set $t \leftarrow t + 1$
    else
      Set $\rho \leftarrow \frac{\rho}{2}$
      Set $t \leftarrow 1$
    end if
  end if
  if $\gamma^k = 0$ **then**
    `Stop; The current solution is optimal`
  else
    $\hat{\theta} \leftarrow$ `applyMHeuristic` $\left(x^k, f, \mathcal{X} \cap \mathcal{F}\right)$
    for all $j \in J$ **do**
      $\pi_j^{k+1} \leftarrow \max \left\{0, \pi_j^k - \left(\rho \frac{\theta\left(\pi^k\right) - \hat{\theta}}{\|\gamma^k\|}\right) \gamma_j^k\right\}$
    end for
    Set $k \leftarrow k + 1$
  end if
until $\frac{\left|f\left(x^k\right) - \theta\left(\pi^k\right)\right|}{\theta\left(\pi^k\right)} \leq \varepsilon$ or $k >$ `iterLimit`

La valeur de $\hat{\theta}$ à l'instruction 27 s'obtient en appliquant une heuristique ou une méta-heuristique au problème primal et non à la relaxation lagrangienne. L'heuristique ou la méta-heuristique est initialisée avec la solution partielle x^k obtenue en résolvant la relaxation lagrangienne. Ainsi, la valeur $\hat{\theta}$ représente la valeur de la meilleure solution réalisable (pour le problème primal) trouvée jusque là.

Pour réussir l'implémentation de l'algorithme du sous-gradient, nous vous recommandons de procéder de façon méthodique. En premier, il vous faut choisir une instance de petite taille qui servira d'instance test. Ensuite, lors du développement il ne faut pas perdre de vue votre but, à savoir, votre code doit être générique et ne doit pas être utilisable uniquement avec l'instance test que vous avez choisie. Vous pouvez, à titre de suggestion, mettre en pratique vos connaissances en modélisation `UML`. Enfin, lors de vos tests, la valeur de la fonction duale (comme vu en cours) est toujours un minorant de la valeur optimale du primal et, par conséquent, de la valeur de toute solution réalisable.

Instructions générales

Tous les sujets traitent de problèmes d'optimisation en variables mixtes. Dans l'ensemble, pour chaque problème il sera demandé d'effectuer les trois expérimentations ci-dessous. Nous entendons par *expérimentation* un ensemble d'expériences de calcul dans un but bien défini. Voici les expérimentations dont il est question plus haut :

Résolution exacte Le but de cette expérimentation est double. En premier, nous vous demanderons de vérifier quelques propriétés sur les différents modèles envisagés. En second, nous vous proposons de comparer, sur les instances fournies, les performances de quelques solveurs (cf. à l'énoncé du sujet) dédiés à la résolution de problèmes d'optimisation en variable mixtes.

Résolution approchée La résolution approchée signifie résoudre le problème à l'aide d'heuristiques ou de méta-heuristiques. Comme pour la résolution exacte, l'objectif de cette partie sera d'identifier une ou plusieurs méthodes approchées performantes pour résoudre le problème.

Algorithme de sous-gradient Dans cette expérimentation vous comparerez l'efficacité d'un algorithme de type sous-gradient (que vous implémenterez) avec le meilleur des solveurs identifié dans la première partie, d'une part. D'autre part, vous analyserez les performances de l'algorithme sous-gradient amendé avec la meilleure des heuristiques ou méta-heuristiques identifiées dans la deuxième partie.

Pour mesurer la qualité d'une solution réalisable obtenue à l'aide d'une méta-heuristiques vous pouvez utiliser la mesure suivante. Notons $\texttt{opt}$ la valeur de la solution optimale du problème et soit $\texttt{opt}^H$ la valeur de la solution trouvée à l'aide d'une méta-heuristique `H`. La qualité de cette dernière peut-être mesurée (pour un problème en minimisation)

$$E = \frac{\texttt{opt}^H - \texttt{opt}}{\texttt{opt}^H} \times 100, \tag{2}$$

à condition que $\texttt{opt}^H$ ne soit pas nulle. Plus le rapport E est petit meilleure est la solution trouvée. Dans le cas où la valeur `opt` n'est pas connue nous pouvons la remplacer par un minorant. Par exemple, la valeur de la fonction duale dans le cas d'une relaxation lagrangienne.

Instances

Des instances (jeux de données) sont fournies pour chaque sujet proposé. Vous les utiliserez pour les diverses expériences de calcul.

La difficulté d'une instance est proportionnelle à sa taille. Donc, nous vous recommandons lors de l'implémentation de l'algorithme de sous-gradient (ou autres tests préliminaires) d'utiliser les petites instances. Puis, une fois votre algorithme validé vous testerez ses performances sur les instances de moyennes et grandes tailles. Pour toutes vos expériences de calcul vous limiterez le temps de calcul à 2 heures.

Délivrables

Vous aurez à livrer deux délivrables. Un rapport (rédigé en LaTeX) dans lequel il sera présenté et analysé les différentes expérimentations demandées. Le second délivrable est constitué des différents codes relatifs aux trois expérimentations. Il sera retourné sous forme d'un dossier. Nous vous suggérons l'organisation minimale suivante de ce dossier :

— `mod` : dossier contenant les modèles **ampl**.

— `mat` : dossier contenant les fichiers **matlab**.

— `sgr` : dossier contenant votre algorithme de sous-gradient.

— `dat` : dossier contenant les jeux de données fournis.

— `rap` : dossier contenant les fichiers sources de votre rapport. Pensez à bien organiser ce dossier (par exemple, les fichiers TEX dans un dossier `tex`, les images et graphiques dans un dossier `img`, ...).

Bien sûr, vous avez tout le loisir pour y ajouter des dossiers autres que ceux mentionnés ci-dessus.

Échéancier

Le projet est à rendre, au plus tard, le 20 **janvier** 2017. L'ensemble des documents le constituant seront à fournir dans une archive `zip` nommée comme suit : `nomprojet-main.zip`. Ce dossier est à envoyer à : `evals.ouzia@gmail.com`

Entre la séance de suivi et la date de remise du projet les questions par courriel sont les bienvenues. Cependant, évitez des questions isomorphes à « Pouvez-vous corriger ceci et cela ... » ou à « Pouvez-vous corriger les erreurs de syntaxe et compléter le code ... ».

Enfin, il sera possible en cas de difficulté majeure de vous recevoir pour vous conseiller (à partir de la rentrée uniquement).

Critères d'évaluation

Les projets seront évalués selon les critères suivants : (i) La pertinence des algorithmes proposés ; (ii) La conduite et l'analyse des expériences de calcul (iii) La qualité du rapport ; (iv) La qualité du code (organisation des fichiers, clarté du code lui-même, ...) ; (v) L'apport personnel (toute initiative personnel est très bien appréciée) ; (vi) La présentation orale. Nous vous recommandons de préparer quelques transparents au travers desquels vous résumerez l'essentiel de vos expérimentations. La présentation durera 30 minutes questions comprises.

Liste des sujets

Projet 1

Le but de ce projet est de résoudre le problème $(\mathcal{P})$ suivant :

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

s.c.

$$\sum_{j=1}^{n} a_{ij}x_{ij} \leq b_i, i \in \{1,\dots,m\}, \tag{3}$$

$$\sum_{i=1}^{m} x_{ij} = 1, j \in \{1,\dots,n\}, \tag{4}$$

$$x \in \{0,1\}^{m\times n}, \tag{5}$$

où :

— $A = (a_{ij})$ et $C = (c_{ij})$ sont des matrices appartenant à $\mathcal{M}_{m\times n}(\mathbb{R})$,

— $b = (b_i)$ est un vecteur appartenant à $\mathbb{R}^m$.

Il s'agit d'un problème d'optimisation en variables entières. Ce problème est **NP-Complet**, en général.

La *relaxation continue* (cf. introduction) du problème $(\mathcal{P})$ est le problème *d'optimisation linéaire* suivant :

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

s.c.

$$\sum_{j=1}^{n} a_{ij}x_{ij} \leq b_i, i \in \{1,\dots,m\},$$

$$\sum_{i=1}^{m} x_{ij} = 1, j \in \{1,\dots,n\},$$

$$x \in [0,1]^{m\times n}.$$

La solution optimale de cette relaxation continue n'est, en général, pas entière. En revanche, sa valeur est toujours un minorant de la valeur d'une solution optimale du problème en variables entières.

Considérons le problème $(\mathcal{M})$ équivalent au problème $(\mathcal{P})$ suivant :

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

s.c.

$$\sum_{j=1, j\neq k}^{n} a_{ij}x_{ij}x_{ik} \leq (b_i - a_{ik})\, x_{ik}, i,k \in \{1,\dots,m\}, \tag{6}$$

$$\sum_{j=1, j\neq k}^{n} a_{ij}x_{ij} - \sum_{j=1, j\neq k}^{n} a_{ij}x_{ij}x_{ik} \leq b_i - b_i x_{ik}, i,k \in \{1,\dots,m\}, \tag{7}$$

$$\sum_{i=1}^{m} x_{ij} = 1, j \in \{1,\dots,n\}, \tag{8}$$

$$x \in \{0,1\}^{m\times n}. \tag{9}$$

Ce modèle $(\mathcal{M})$ s'obtient à partir du modèle $(\mathcal{P})$ en multipliant les contraintes (3) par x_{ik} et $1 - x_{ik}$ et ce pour toutes les paires d'indices (i,k). Cette manipulation est valide car x_{ik} et $1 - x_{ik}$ sont positives et que x_{ik}^2 est égal à x_{ik} (car x_{ik}

est une variable binaire). Donc, l'ensemble des solutions réalisables du problème $(\mathcal{P})$ n'est pas altéré. En revanche, le modèle $(\mathcal{M})$ ainsi obtenu est plus difficile à résoudre que le modèle originel. Cette difficulté est essentiellement due aux nombres de variables et contraintes du nouveau modèle $(\mathcal{M})$. Nous verrons que l'intérêt d'un tel modèle est dans sa relaxation continue.

Enfin, considérons une linéarisation possible du modèle $(\mathcal{M})$. Cette linéarisation consiste : (i) à remplacer les termes non linéaires $x_{ij}x_{ik}$ par de nouvelles variables notées z^i_{jk} ; (ii) Puis, à ajouter des contraintes, dites valides, liant les nouvelles variables et les variables originelles. Nous considérerons seulement les contraintes suivantes :

$$\begin{aligned} z^i_{jk} &\geq 0, \forall i,j,k,\\ z^i_{jk} &\leq x_{ij},\ \forall i,j,k,\\ z^i_{jk} &\leq x_{ik},\ \forall i,j,k,\\ x_{ik}+x_{ij}-z^i_{jk} &\leq 1,\ \forall i,j,k. \end{aligned}$$

La validité des trois premiers groupes de contraintes ci-dessus est claire. Pour se convaincre de la validité du dernier groupe de contraintes, il suffit de voir que si les deux variables binaires x_{ij} et x_{ik} valent 1 alors cette dernière contrainte impose que la valeur de la variable z^i_{jk} soit égale aussi à 1.

Ainsi, le modèle linéarisé (que nous noterons $(\mathcal{L})$) du modèle $(\mathcal{M})$ est le modèle suivant :

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

s.c.

$$\sum_{j=1}^{n} a_{ij}z^i_{jk} \leq (b_i - a_{ik})\, x_{ik}, i,k \in \{1,\ldots,m\}, \tag{10}$$

$$\sum_{j=1,j\neq k}^{n} a_{ij}x_{ij} - \sum_{j=1,j\neq k}^{n} a_{ij}z^i_{jk} \leq b_i - b_i x_{ik}, i,k \in \{1,\ldots,m\}, \tag{11}$$

$$\sum_{i=1}^{m} x_{ij} = 1, j \in \{1,\ldots,n\}, \tag{12}$$

$$z^i_{jk} \leq x_{ij},\ \forall i,j,k, \tag{13}$$

$$z^i_{jk} \leq x_{ik},\ \forall i,j,k, \tag{14}$$

$$x_{ik}+x_{ij}-z^i_{jk} \leq 1,\ \forall i,j,k, \tag{15}$$

$$x \in \{0,1\}^{m\times n}, z \geq 0. \tag{16}$$

Les questions de chacune des trois parties ci-dessous sont indicatives. Ne pas hésiter à proposer des expériences de calcul complémentaires pour étayer vos observations.

Partie A : Résolution exacte

Les premières expériences suggérées dans cette partie ont pour objectif de vérifier quelques propriétés théoriques liant les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Ces expériences seront à effectuer sur quelques instances de petites tailles.

1. Vérifier l'équivalence des deux modèle $(\mathcal{L})$ et $(\mathcal{P})$.
2. Vérifier que la valeur optimale de la relaxation continue du modèle $(\mathcal{L})$ est au moins aussi bonne que la valeur optimale de la relaxation continue du problème originel $(\mathcal{P})$. Autrement, si nous appelons $\texttt{opt}^P$ et $\texttt{opt}^L$ les valeurs optimales des relaxations continues des problèmes $(\mathcal{P})$ et $(\mathcal{L})$ respectivement alors vous obtiendrez, sur toutes les instances testées :

   $$\texttt{opt}^P \leq \texttt{opt}^L.$$

   Et, bien sûr, la valeur $\texttt{opt}^L$ est inférieure ou égale à la valeur de la solution optimale entière de l'instance.

Les expériences suivantes ont pour objectif de comparer les performances (temps de calcul et valeur de la solution trouvée) de quelques solveurs pour résoudre les modèles $(\mathcal{P})$, $(\mathcal{L})$ et $(\mathcal{M})$. Plus de détails vous seront donnés lors de la séance de suivi au sujet des solveurs choisis.

3. Comparer les performances des solveurs `intlinprog`, `cplex`, `gurobi` et `xpress` sur les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Pour ces expériences vous utiliserez `AMPL` (version commerciale de `gmpl`).
4. Comparer les performances des deux solveurs `baron` et `knitro` pour résoudre le modèle $(\mathcal{M})$. Il s'agit d'un problème non linéaire en variables entières. Donc, un problème assez difficile.

Partie B : Résolution approchée

Pour cette partie, l'objectif est d'identifier un *bonne méta-heuristique* pour résoudre les problèmes $(\mathcal{P})$ et $(\mathcal{L})$. Une méta-heuristique sera qualifiée de bonne si elle retourne une solution réalisable de bonne valeur (pour le problème $(\mathcal{P})$) en un temps réduit.

5. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre le problème $(\mathcal{P})$.
6. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (10).
7. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (11).

Vous pouvez tester les performances d'autres méta-heuristiques. Cependant, ne perdez pas beaucoup de temps dans les tracas informatiques : installation, désinstallation, libraires ...

Partie C : Algorithme de sous-gradient

Cette dernière partie à pour objectif de résoudre le problème $(\mathcal{L})$ à l'aide de la technique de relaxation lagrangienne.

Il existe plusieurs relaxations lagrangiennes du problème $(\mathcal{L})$. Ces relaxations se distinguent par les contraintes dualisées. Nous vous proposons d'en tester deux de ces relaxations. Vous pouvez, bien sûr, en comparer d'autres.

Vous utiliserez dans l'algorithme de sous-gradient la meilleure des deux heuristiques identifiée dans la partie précédente.

8. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (10). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
9. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (11). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
10. Commenter vos résultats.

Instances

Pour ce projet, cinq jeux de données $(1,\ldots,5)$ vous sont proposés. Dans chaque jeu vous avez des instances de différentes tailles. Les instances du jeu i sont plus difficiles que celles du jeu $i-1$, pour i dans $\{2,\ldots 5\}$.

Le format utilisé dans chaque fichier de données est le suivant :

☞ La première ligne contient les valeur des entiers m et n, dans cet ordre.

☞ Suit les valeurs $C = (c_{ij})$. Chaque ligne de la matrice C est présentée sur 9 lignes et 12 colonnes. La dernière ligne contenant seulement 4 éléments.

☞ Puis, les valeurs de la matrice $A = (a_{ij})$. Les lignes de la matrice A sont présentées de la même manière que les lignes de la matrice C.

☞ Enfin, les valeurs (b_i) en 9 lignes et 12 colonnes.

Projet 2

Le but de ce projet est de résoudre le problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\qquad \sum_{i=1}^{m} x_{ij} \geq A_j,\ j\in\{1,\dots,n\}, && (17)\\
&\qquad \sum_{j=1}^{n} x_{ij} \leq M_i,\ i\in\{1,\dots,m\}, && (18)\\
&\qquad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j, && (19)\\
&\qquad y\in\{0,1\}^{m\times n}, x\geq 0, && (20)
\end{aligned}$$

où :

— $c=\left(c_{ij}\right)$, $a=\left(a_{ij}\right)$ et $m=\left(m_{ij}\right)$ sont des matrices appartenant à $\mathcal{M}_{m\times n}(\mathbb{R})$,

— $A=\left(A_j\right)$ et $M=\left(M_i\right)$ sont deux vecteurs appartenant à $\mathbb{R}^m$ et $\mathbb{R}^n$, respectivement.

Il s'agit d'un problème d'optimisation en variables entières. Ce problème est **NP-Complet**, en général.

La *relaxation continue* (cf. introduction) du problème $(\mathcal{P})$ est le problème *d'optimisation linéaire* suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\qquad \sum_{i=1}^{m} x_{ij} \geq A_j,\ j\in\{1,\dots,n\},\\
&\qquad \sum_{j=1}^{n} x_{ij} \leq M_i,\ i\in\{1,\dots,m\},\\
&\qquad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j,\\
&\qquad y\in[0,1]^{m\times n}, x\geq 0.
\end{aligned}$$

La solution optimale de cette relaxation continue n'est, en général, pas entière. En revanche, sa valeur est toujours un minorant de la valeur d'une solution optimale du problème en variables entières.

Considérons le problème $(\mathcal{M})$ équivalent au problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\qquad \sum_{i=1}^{m} x_{ij}y_{kj} \geq A_j y_{kj},\ \forall j,k, && (21)\\
&\qquad \sum_{i=1}^{m} x_{ij} - \sum_{j=1}^{n} x_{ij}y_{kj} \geq A_j - A_j y_{kj},\ \forall j,k, && (22)\\
&\qquad \sum_{j=1}^{n} x_{ij} \leq M_i,\ i\in\{1,\dots,m\}, && (23)\\
&\qquad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j, && (24)\\
&\qquad y\in\{0,1\}^{m\times n}, x\geq 0. && (25)
\end{aligned}$$

Ce modèle $(\mathcal{M})$ s'obtient à partir du modèle $(\mathcal{P})$ en multipliant les contraintes (17) par y_{kj} et $1 - y_{kj}$ et ce pour toutes les paires d'indices (k, j). Cette manipulation est valide car y_{kj} et $1 - y_{kj}$ sont positives. Donc, l'ensemble des solutions réalisables du problème $(\mathcal{P})$ n'est pas altéré. En revanche, le modèle $(\mathcal{M})$ ainsi obtenu est plus difficile à résoudre que le modèle originel. Cette difficulté est essentiellement due aux nombres de variables et contraintes du nouveau modèle $(\mathcal{M})$. Nous verrons que l'intérêt d'un tel modèle est dans sa relaxation continue.

Enfin, considérons une linéarisation possible du modèle $(\mathcal{M})$. Cette linéarisation consiste : (i) à remplacer les termes non linéaires $x_{ij}y_{kj}$ par de nouvelles variables notées z_{ik}^j ; (ii) Puis, à ajouter des contraintes, dites valides, liant les nouvelles variables et les variables originelles. Nous considérerons seulement les contraintes suivantes :

$$\begin{aligned} z_{ik}^j &\geq 0, \forall i, j, k, \\ z_{ik}^j &\leq y_{kj}, \ \forall i, j, k, \\ z_{ik}^j &\leq x_{ij}, \ \forall i, j, k, \\ x_{ij} + y_{kj} - z_{ik}^j &\leq 1, \ \forall i, j, k. \end{aligned}$$

La validité des trois premiers groupes de contraintes ci-dessus est claire. Pour se convaincre de la validité du dernier groupe de contraintes, il suffit de voir que si les deux variables binaires x_{ij} et y_{kj} valent 1 alors cette dernière contrainte impose que la valeur de la variable z_{ik}^j soit égale aussi à 1.

Ainsi, le modèle linéarisé (que nous noterons $(\mathcal{L})$) du modèle $(\mathcal{M})$ est le modèle suivant :

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} y_{ij} \left(a_{ij} + c_{ij} x_{ij} \right)$$

s.c.

$$\sum_{i=1}^{m} z_{ik}^j \geq A_j y_{kj}, \ \forall j, k, \tag{26}$$

$$\sum_{i=1}^{m} x_{ij} - \sum_{i=1}^{m} z_{ik}^j \geq A_j - A_j y_{kj}, \ \forall j, k, \tag{27}$$

$$\sum_{j=1}^{n} x_{ij} \leq M_i, \ i \in \{1, \ldots, m\}, \tag{28}$$

$$m_{ij} x_{ij} \leq y_{ij} \leq M_i x_{ij}, \ \forall i, j, \tag{29}$$

$$z_{ik}^j \leq y_{kj}, \ \forall i, j, k, \tag{30}$$

$$z_{ik}^j \leq x_{ij}, \ \forall i, j, k, \tag{31}$$

$$x_{ij} + y_{kj} - z_{ik}^j \leq 1, \ \forall i, j, k, \tag{32}$$

$$y \in \{0, 1\}^{m \times n}, x, z \geq 0. \tag{33}$$

Les questions de chacune des trois parties ci-dessous sont indicatives. Ne pas hésiter à proposer des expériences de calcul complémentaires pour étayer vos observations.

Partie A : Résolution exacte

Les premières expériences suggérées dans cette première partie ont pour objectif de vérifier quelques propriétés théoriques liant les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Ces expériences seront à effectuer sur quelques instances de petites tailles.

1. Vérifier l'équivalence des deux modèle $(\mathcal{L})$ et $(\mathcal{P})$.
2. Vérifier que la valeur optimale de la relaxation continue du modèle $(\mathcal{L})$ est au moins aussi bonne que la valeur optimale de la relaxation continue du problème originel $(\mathcal{P})$. Autrement, si nous appelons $\texttt{opt}^P$ et $\texttt{opt}^L$ les valeurs optimales des relaxations continues des problèmes $(\mathcal{P})$ et $(\mathcal{L})$ respectivement alors vous obtiendrez, sur toutes les instances testées :

   $$\texttt{opt}^P \leq \texttt{opt}^L.$$

   Et, bien sûr, la valeur $\texttt{opt}^L$ est inférieure ou égale à la valeur de la solution optimale entière de l'instance.

Les expériences suivantes ont pour objectif de comparer les performances (temps de calcul et valeur de la solution trouvée) de quelques solveurs pour résoudre les modèles $(\mathcal{P})$, $(\mathcal{L})$ et $(\mathcal{M})$. Plus de détails vous seront donnés lors de la séance de suivi au sujet des solveurs choisis.

3. Comparer les performances des solveurs `intlinprog`, `cplex`, `gurobi` et `xpress` sur les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Pour ces expériences vous utiliserez `AMPL` (version commerciale de `gmpl`).
4. Comparer les performances des deux solveurs `baron` et `knitro` pour résoudre le modèle $(\mathcal{M})$. Il s'agit d'un problème non linéaire en variables entières. Donc, un problème assez difficile.

Partie B : Résolution approchée

Pour cette partie, l'objectif est d'identifier un *bonne méta-heuristique* pour résoudre les problèmes $(\mathcal{P})$ et $(\mathcal{L})$. Une méta-heuristique sera qualifiée de bonne si elle retourne une solution réalisable de bonne valeur (pour le problème $(\mathcal{P})$) en un temps réduit.

5. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre le problème $(\mathcal{P})$.
6. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (26).
7. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (27).

Vous pouvez tester les performances d'autres méta-heuristiques. Cependant, ne perdez pas beaucoup de temps dans les tracas informatiques : installation, désinstallation, libraires ...

Partie C : Algorithme de sous-gradient

Cette dernière partie à pour objectif de résoudre le problème $(\mathcal{L})$ à l'aide de la technique de relaxation lagrangienne.

Il existe plusieurs relaxations lagrangiennes du problème $(\mathcal{L})$. Ces relaxations se distinguent par les contraintes dualisées. Nous vous proposons d'en comparer deux de ces relaxations. Vous pouvez, bien sûr, en tester d'autres.

Vous utiliserez dans l'algorithme de sous-gradient la meilleure des deux heuristiques identifiée dans la partie précédente.

8. Résoudre à l'aide d'un algorithme de sous-gradient la relaxations continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (26). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
9. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (27). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
10. Commenter vos résultats.

Instances

Quatre jeux de données vous sont fournis. Les données dans chaque fichier sont explicites. Les valeurs des paramètres du problème sont organisées en tableaux ou matrices à la Matlab. Les noms des paramètres sont légèrement différents de ceux de l'énoncé (e.g., la matrice $\left(a_{ij}\right)$ est nommée a, le vecteur M est nommé MM ...).

Projet 3

Le but de ce projet est de résoudre le problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m} c_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} x_{ij} \\
&\text{s.c.} \\
&\quad \sum_{j=1}^{n} a_{ij} x_{ij} \leq b_i y_i, i \in \{1, \ldots, m\}, && (34)\\
&\quad \sum_{i=1}^{m} x_{ij} = 1, j \in \{1, \ldots, n\}, && (35)\\
&\quad x \in \{0,1\}^{m\times n}, y \in \{0,1\}^{m}, && (36)
\end{aligned}$$

où :

- $A = (a_{ij})$ et $D = (d_{ij})$ sont des matrices appartenant à $\mathcal{M}_{m\times n}(\mathbb{R})$,
- $b = (b_i)$ et $c = (c_i)$ sont deux vecteurs appartenant à $\mathbb{R}^m$.

Il s'agit d'un problème d'optimisation en variables entières. Ce problème est **NP-Complet**, en général.

La *relaxation continue* (cf. introduction) du problème $(\mathcal{P})$ est le problème *d'optimisation linéaire* suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m} c_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} x_{ij} \\
&\text{s.c.} \\
&\quad \sum_{j=1}^{n} a_{ij} x_{ij} \leq b_i y_i, i \in \{1, \ldots, m\}, \\
&\quad \sum_{i=1}^{m} x_{ij} = 1, j \in \{1, \ldots, n\}, \\
&\quad x \in [0,1]^{m\times n}, y \in [0,1]^{m}.
\end{aligned}$$

La solution optimale de cette relaxation continue n'est, en général, pas entière. En revanche, sa valeur est toujours un minorant de la valeur d'une solution optimale du problème en variables entières.

Considérons le problème $(\mathcal{M})$ équivalent au problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m} c_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} x_{ij} \\
&\text{s.c.} \\
&\quad \sum_{j=1}^{n} a_{ij} x_{ij} y_i \leq b_i y_i, i \in \{1, \ldots, m\}, && (37)\\
&\quad \sum_{j=1}^{n} a_{ij} x_{ij} - \sum_{j=1}^{n} a_{ij} x_{ij} y_i \leq 0, i \in \{1, \ldots, m\}, && (38)\\
&\quad \sum_{i=1}^{m} x_{ij} = 1, j \in \{1, \ldots, n\}, && (39)\\
&\quad x \in \{0,1\}^{m\times n}, y \in \{0,1\}^{m}. && (40)
\end{aligned}$$

Ce modèle $(\mathcal{M})$ s'obtient à partir du modèle $(\mathcal{P})$ en multipliant les contraintes (34) par y_i et $1 - y_i$ et ce pour toutes les valeurs possible de l'indices i. Cette manipulation est valide car y_i et $1 - y_i$ sont positives. Donc, l'ensemble des solutions réalisables du problème $(\mathcal{P})$ n'est pas altéré. En revanche, le modèle $(\mathcal{M})$ ainsi obtenu est plus difficile à résoudre que le modèle originel. Cette difficulté est essentiellement due aux nombres de variables et contraintes du nouveau modèle $(\mathcal{M})$. Nous verrons que l'intérêt d'un tel modèle réside dans sa relaxation continue.

Enfin, considérons une linéarisation possible du modèle $(\mathcal{M})$. Cette linéarisation consiste : (i) à remplacer les termes non linéaires $x_{ij}y_i$ par de nouvelles variables notées z_{ij} ; (ii) Puis, à ajouter des contraintes, dites valides, liant les nouvelles variables et les variables originelles. Nous considérerons seulement les contraintes suivantes :

$$\begin{aligned} z_{ij} &\geq 0, \forall i,j, \\ z_{ij} &\leq x_{ij},\ \forall i,j, \\ z_{ij} &\leq y_i,\ \forall i,j, \\ x_{ij} + y_i - z_{ij} &\leq 1,\ \forall i,j. \end{aligned}$$

La validité des trois premiers groupes de contraintes ci-dessus est claire. Pour se convaincre de la validité du dernier groupe de contraintes, il suffit de voir que si les deux variables binaires x_{ij} et y_i valent 1 alors cette dernière contrainte impose que la valeur de la variable z_{ij} soit égale aussi à 1.

Ainsi, le modèle linéarisé (que nous noterons $(\mathcal{L})$) du modèle $(\mathcal{M})$ est le modèle suivant :

$$\min \sum_{i=1}^{m} c_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij}x_{ij}$$

s.c.

$$\sum_{j=1}^{n} a_{ij}z_{ij} \leq b_i y_i, i \in \{1,\ldots,m\}, \tag{41}$$

$$\sum_{j=1}^{n} a_{ij}x_{ij} - \sum_{j=1}^{n} a_{ij}z_{ij} \leq 0, i \in \{1,\ldots,m\}, \tag{42}$$

$$\sum_{i=1}^{m} x_{ij} = 1, j \in \{1,\ldots,n\}, \tag{43}$$

$$z_{ij} \leq x_{ij},\ \forall i,j, \tag{44}$$

$$z_{ij} \leq y_i,\ \forall i,j, \tag{45}$$

$$x_{ij} + y_i - z_{ij} \leq 1,\ \forall i,j, \tag{46}$$

$$x \in \{0,1\}^{m\times n}, y \in \{0,1\}^{m}, z \geq 0. \tag{47}$$

Les questions de chacune des trois parties ci-dessous sont indicatives. Ne pas hésiter à proposer des expériences de calcul complémentaires pour étayer vos observations.

Partie A : Résolution exacte

Les premières expériences suggérées dans cette première partie ont pour objectif de vérifier quelques propriétés théoriques liant les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Ces expériences seront à effectuer sur quelques instances de petites tailles.

1. Vérifier l'équivalence des deux modèle $(\mathcal{L})$ et $(\mathcal{P})$.
2. Vérifier que la valeur optimale de la relaxation continue du modèle $(\mathcal{L})$ est au moins aussi bonne que la valeur optimale de la relaxation continue du problème originel $(\mathcal{P})$. Autrement, si nous appelons $\texttt{opt}^P$ et $\texttt{opt}^L$ les valeurs optimales des relaxations continues des problèmes $(\mathcal{P})$ et $(\mathcal{L})$ respectivement alors vous obtiendrez, sur toutes les instances testées :

   $$\texttt{opt}^P \leq \texttt{opt}^L.$$

   Et, bien sûr, la valeur $\texttt{opt}^L$ est inférieure ou égale à la valeur de la solution optimale entière de l'instance.

Les expériences suivantes ont pour objectif de comparer les performances (temps de calcul et valeur de la solution trouvée) de quelques solveurs pour résoudre les modèles $(\mathcal{P})$, $(\mathcal{L})$ et $(\mathcal{M})$. Plus de détails vous seront donnés lors de la séance de suivi au sujet des solveurs choisis.

3. Comparer les performances des solveurs `intlinprog`, `cplex`, `gurobi` et `xpress` sur les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Pour ces expériences vous utiliserez `AMPL` (version commerciale de `gmpl`).
4. Comparer les performances des deux solveurs `baron` et `knitro` pour résoudre le modèle $(\mathcal{M})$. Il s'agit d'un problème non linéaire en variables entières. Donc, un problème assez difficile.

Partie B : Résolution approchée

Pour cette partie, l'objectif est d'identifier un *bonne méta-heuristique* pour résoudre les problèmes $(\mathcal{P})$ et $(\mathcal{L})$. Une méta-heuristique sera qualifiée de bonne si elle retourne une solution réalisable de bonne valeur (pour le problème $(\mathcal{P})$) en un temps réduit.

5. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre le problème $(\mathcal{P})$.
6. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (41).
7. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (42).

Vous pouvez tester les performances d'autres méta-heuristiques. Cependant, ne perdez pas beaucoup de temps dans les tracs informatiques : installation, désinstallation, libraires ....

Partie C : Algorithme de sous-gradient

Cette dernière partie à pour objectif de résoudre le problème $(\mathcal{L})$ à l'aide de la technique de relaxation lagrangienne.

Il existe plusieurs relaxations lagrangiennes du problème $(\mathcal{L})$. Ces relaxations se distinguent par les contraintes dualisées. Nous vous proposons d'en comparer deux de ces relaxations. Vous pouvez, bien sûr, en tester d'autres.

Vous utiliserez dans l'algorithme de sous-gradient la meilleure des deux heuristiques identifiée dans la partie précédente.

8. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (41) et (42). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
9. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (43). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
10. Commenter vos résultats.

Instances

Pour ce projet, cinq jeux de données $(1, \ldots, 5)$ vous sont proposés. Dans chaque jeu vous avez des instances de même taille. Les instances du jeu i sont plus difficiles que celles du jeu $i-1$, pour i dans $\{2, \ldots 5\}$.

Le format utilisé dans chaque fichier de données est le suivant :

☞ La première ligne contient, dans l'ordre, les valeurs de m, n, c et b où :

$$c_i = c, \ \forall i,$$
$$b_i = b, \ \forall i.$$

☞ Les lignes suivantes respectent le format suivant :

$$i, \ j, \ d_{ij}, \ a_{ij}.$$

Projet 4

Le but de ce projet est de résoudre le problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\quad \sum_{i=1}^{m} x_{ij} \geq A_j,\ j\in\{1,\ldots,n\}\,, && (48)\\
&\quad \sum_{j=1}^{n} x_{ij} \leq M_i,\ i\in\{1,\ldots,m\}\,, && (49)\\
&\quad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j, && (50)\\
&\quad y\in\{0,1\}^{m\times n}, x\geq 0, && (51)
\end{aligned}$$

où :

— $c=(c_{ij})$, $a=(a_{ij})$ et $m=(m_{ij})$ sont des matrices appartenant à $\mathcal{M}_{m\times n}(\mathbb{R})$,

— $A=(A_j)$ et $M=(M_i)$ sont deux vecteurs appartenant à $\mathbb{R}^m$ et $\mathbb{R}^n$, respectivement.

Il s'agit d'un problème d'optimisation en variables entières. Ce problème est **NP-Complet**, en général.

La *relaxation continue* (cf. introduction) du problème $(\mathcal{P})$ est le problème *d'optimisation linéaire* suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\quad \sum_{i=1}^{m} x_{ij} \geq A_j,\ j\in\{1,\ldots,n\}\,,\\
&\quad \sum_{j=1}^{n} x_{ij} \leq M_i,\ i\in\{1,\ldots,m\}\,,\\
&\quad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j,\\
&\quad y\in[0,1]^{m\times n}, x\geq 0.
\end{aligned}$$

La solution optimale de cette relaxation continue n'est, en général, pas entière. En revanche, sa valeur est toujours un minorant de la valeur d'une solution optimale du problème en variables entières.

Considérons le problème $(\mathcal{M})$ équivalent au problème $(\mathcal{P})$ suivant :

$$\begin{aligned}
&\min \sum_{i=1}^{m}\sum_{j=1}^{n} y_{ij}\left(a_{ij}+c_{ij}x_{ij}\right)\\
&\text{s.c.}\\
&\quad \sum_{i=1}^{m} x_{ij} \geq A_j,\ j\in\{1,\ldots,n\}\,, && (52)\\
&\quad \sum_{j=1}^{n} x_{ij}y_{ik} \leq M_i y_{ik},\ \forall i,k, && (53)\\
&\quad \sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n} x_{ij}y_{ik} \leq M_i - M_i y_{ik},\ \forall i,k, && (54)\\
&\quad m_{ij}y_{ij} \leq x_{ij} \leq M_i y_{ij},\ \forall i,j, && (55)\\
&\quad y\in\{0,1\}^{m\times n}, x\geq 0. && (56)
\end{aligned}$$

Ce modèle $(\mathcal{M})$ s'obtient à partir du modèle $(\mathcal{P})$ en multipliant les contraintes (49) par y_{ik} et $1 - y_{ik}$ et ce pour toutes les paires d'indices (i, k). Cette manipulation est valide car y_{ik} et $1 - y_{ik}$ sont positives. Donc, l'ensemble des solutions réalisables du problème $(\mathcal{P})$ n'est pas altéré. En revanche, le modèle $(\mathcal{M})$ ainsi obtenu est plus difficile à résoudre que le modèle originel. Cette difficulté est essentiellement due aux nombres de variables et contraintes du nouveau modèle $(\mathcal{M})$. Nous verrons que l'intérêt d'un tel modèle est dans sa relaxation continue.

Enfin, considérons une linéarisation possible du modèle $(\mathcal{M})$. Cette linéarisation consiste : (i) à remplacer les termes non linéaires $x_{ij}y_{ik}$ par de nouvelles variables notées z^i_{jk} ; (ii) Puis, à ajouter des contraintes, dites valides, liant les nouvelles variables et les variables originelles. Nous considérerons seulement les contraintes suivantes :

$$\begin{aligned}
z^i_{jk} &\geq 0, \forall i, j, k, \\
z^i_{jk} &\leq x_{ij}, \ \forall i, j, k, \\
z^i_{jk} &\leq y_{ik}, \ \forall i, j, k, \\
x_{ij} + y_{ik} - z^i_{jk} &\leq 1, \ \forall i, j, k.
\end{aligned}$$

La validité des trois premiers groupes de contraintes ci-dessus est claire. Pour se convaincre de la validité du dernier groupe de contraintes, il suffit de voir que si les deux variables binaires x_{ij} et y_{ik} valent 1 alors cette dernière contrainte impose que la valeur de la variable z^i_{jk} soit égale aussi à 1.

Ainsi, le modèle linéarisé (que nous noterons $(\mathcal{L})$) du modèle $(\mathcal{M})$ est le modèle suivant :

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} y_{ij} \left(a_{ij} + c_{ij} x_{ij} \right)$$

s.c.

$$\sum_{i=1}^{m} x_{ij} \geq A_j, \ j \in \{1, \ldots, n\}, \tag{57}$$

$$\sum_{j=1}^{n} z^i_{jk} \leq M_i y_{ik}, \ \forall i, k, \tag{58}$$

$$\sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n} z^i_{jk} \leq M_i - M_i y_{ik}, \ \forall i, k, \tag{59}$$

$$m_{ij} x_{ij} \leq y_{ij} \leq M_i x_{ij}, \ \forall i, j, \tag{60}$$

$$z^i_{jk} \leq x_{ij}, \ \forall i, j, k, \tag{61}$$

$$z^i_{jk} \leq y_{ik}, \ \forall i, j, k, \tag{62}$$

$$x_{ij} + y_{ik} - z^i_{jk} \leq 1, \ \forall i, j, k, \tag{63}$$

$$y \in \{0, 1\}^{m \times n}, x, z \geq 0. \tag{64}$$

Les questions de chacune des trois parties ci-dessous sont indicatives. Ne pas hésiter à proposer des expériences de calcul complémentaires pour étayer vos observations.

Partie A : Résolution exacte

Les premières expériences suggérées dans cette première partie ont pour objectif de vérifier quelques propriétés théoriques liant les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Ces expériences seront à effectuer sur quelques instances de petites tailles.

1. Vérifier l'équivalence des deux modèle $(\mathcal{L})$ et $(\mathcal{P})$.
2. Vérifier que la valeur optimale de la relaxation continue du modèle $(\mathcal{L})$ est au moins aussi bonne que la valeur optimale de la relaxation continue du problème originel $(\mathcal{P})$. Autrement, si nous appelons $\mathtt{opt}^P$ et $\mathtt{opt}^L$ les valeurs optimales des relaxations continues des problèmes $(\mathcal{P})$ et $(\mathcal{L})$ respectivement alors vous obtiendrez, sur toutes les instances testées :

$$\mathtt{opt}^P \leq \mathtt{opt}^L.$$

Et, bien sûr, la valeur $\mathtt{opt}^L$ est inférieure ou égale à la valeur de la solution optimale entière de l'instance.

Les expériences suivantes ont pour objectif de comparer les performances (temps de calcul et valeur de la solution trouvée) de quelques solveurs pour résoudre les modèles $(\mathcal{P})$, $(\mathcal{L})$ et $(\mathcal{M})$. Plus de détails vous seront donnés lors de la séance de suivi au sujet des solveurs choisis.

3. Comparer les performances des solveurs `intlinprog`, `cplex`, `gurobi` et `xpress` sur les deux modèles $(\mathcal{P})$ et $(\mathcal{L})$. Pour ces expériences vous utiliserez `AMPL` (version commerciale de `gmpl`).
4. Comparer les performances des deux solveurs `baron` et `knitro` pour résoudre le modèle $(\mathcal{M})$. Il s'agit d'un problème non linéaire en variables entières. Donc, un problème assez difficile.

Partie B : Résolution approchée

Pour cette partie, l'objectif est d'identifier un *bonne méta-heuristique* pour résoudre les problèmes $(\mathcal{P})$ et $(\mathcal{L})$. Une méta-heuristique sera qualifiée de bonne si elle retourne une solution réalisable de bonne valeur (pour le problème $(\mathcal{P})$) en un temps réduit.

5. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre le problème $(\mathcal{P})$.
6. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (58).
7. Comparer les performances des deux méta-heuristiques `ga` et `patternsearch` pour résoudre $(\mathcal{P})$ en les initialisant avec la solution de la relaxation continue du problème $(\mathcal{L})$ sans les contraintes (59).

Vous pouvez tester les performances d'autres méta-heuristiques. Cependant, ne perdez pas beaucoup de temps dans les tracas informatiques : installation, désinstallation, libraires ...

Partie C : Algorithme de sous-gradient

Cette dernière partie à pour objectif de résoudre le problème $(\mathcal{L})$ à l'aide de la technique de relaxation lagrangienne.

Il existe plusieurs relaxations lagrangiennes du problème $(\mathcal{L})$. Ces relaxations se distinguent par les contraintes dualisées. Nous vous proposons d'en comparer deux de ces relaxations. Vous pouvez, bien sûr, en tester d'autres.

Vous utiliserez dans l'algorithme de sous-gradient la meilleure des deux heuristiques identifiée dans la partie précédente.

8. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (58). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
9. Résoudre à l'aide d'un algorithme de sous-gradient la relaxation continue du problème $(\mathcal{L})$ en dualisant seulement les contraintes (59). Vous utiliserez la meilleure méta-heuristique pour identifier une solution réalisable entière du problème $(\mathcal{P})$.
10. Commenter vos résultats.

Instances

Quatre jeux de données vous sont fournis. Les données dans chaque fichier sont explicites. Les valeurs des paramètres du problème sont organisées en tableaux ou matrices à la Matlab. Les noms des paramètres sont légèrement différents de ceux de l'énoncé (e.g., la matrice $\left(a_{ij}\right)$ est nommée a, le vecteur M est nommé MM ...).