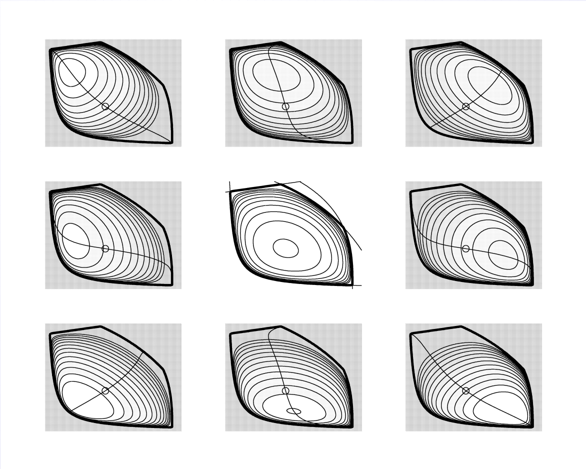
Les multiples facettes de l'optimisation

François Glineur
UCL/FSA/INMA & CORE, Chaire Tractebel
glineur@core.ucl.ac.be



Colloquium MAPA

13 mars 2003

Motivation

Modélisation et aide à la décision

Aider à choisir la meilleure décision

$$\left.\begin{array}{rcl}\text{Décision} & \leftrightarrow & \text{vecteur de variables}\\ \text{Meilleure} & \leftrightarrow & \text{fonction objectif}\\ \text{Contraintes} & \leftrightarrow & \text{domaine admissible}\end{array}\right\} \Rightarrow \text{Optimisation}$$

Utilisation

- ◇ Nombreuses applications en pratique
- ◇ Méthodes de résolution efficaces en pratique
- ◇ Modélisation et résolution de modèles de grande taille

Plan

Introduction

- ◇ Motivation et applications
- ◇ Formulation et taxonomie

Optimisation[1] linéaire

- ◇ Trois problèmes typiques
- ◇ Algorithmes de complexité
- ◇ Dualité

Optimisation non-linéaire et convexe

- ◇ Motivation
- ◇ Optimisation conique

[1] (anciennement) programmation

Introduction

Applications

- Planification, gestion et ordonnancement
  Production, horaires, composition d'équipages, etc.
- Design et conception
  Dimensionnement, optimisation de structures, de réseaux
- Économie et finances
  Choix de portefeuille, calcul d'équilibres
- Localisation et transport
  Placement de dépôts, de circuits intégrés, tournées
- Et beaucoup d'autres ...

Les deux visages de l'optimisation

- Modélisation

  Traduction en langage mathématique du problème (tâche plus délicate qu'il n'y paraît)

$$\Updownarrow$$

Formulation d'un problème d'optimisation

$$\Updownarrow$$

- Résolution

  Développement et implémentation d'algorithmes de résolution en *théorie* et en *pratique*

Relation étroite

- ◇ Formuler des modèles que l'on sait résoudre
- ◇ Développer des méthodes applicables à des modèles réalistes

Formulation classique

$$\min_{x \in \mathbb{R}^n} f(x) \text{ tel que } x \in \mathcal{D} \subseteq \mathbb{R}^n$$

(dimension finie). Souvent on utilise

$$\mathcal{D} = \{x \in \mathbb{R}^n | f_i(x) \leq 0 \text{ et } g_j(x) = 0 \text{ pour } i \in I, j \in J\}$$

Une taxonomie

- ◇ Problème déterministe* ou stochastique
- ◇ Données précises* ou incertaines/floues (robustesse)
- ◇ Problème mono* ou multi-critère
- ◇ Problème contraint ou non-contraint
- ◇ Caractéristiques sous forme analytique* ou boîtes noires
- ◇ Fonctions continues ou non, différentiables ou non
- ◇ Fonctions analytiques, polynomiales, quadratiques, linéaires
- ◇ Variables continues ou discrètes

Changer de catégorie: parfois possible via *reformulation*

Pour la suite: caractéristiques *

Optimisation linéaire: trois exemples

A. Problème du régime

Soit un ensemble de denrées pour lesquelles on connaît

- ◇ La quantité de calories, protéines, glucides, lipides, vitamines renfermée par unité de poids
- ◇ Le prix par unité de poids

Étant données les recommandations des nutritionnistes en matières de besoins journaliers en protéines, glucides, etc, établir le régime idéal, càd rencontrant ces besoins journaliers au moindre coût

Formulation

- Indice i pour les denrées ($1 \leq i \leq n$)
- Indice j pour composants nutritionnels ($1 \leq j \leq m$)
- Données (par unité de poids) :

  $c_i \longrightarrow$ prix de la denrée i,

  $a_{ji} \longrightarrow$ quantité du composant j dans la denrée i,

  b_j besoin journalier en composant j
- Inconnues:

  quantité x_i de la denrée i intervenant dans le régime

Formulation (suite)

Il s'agit d'un problème (programme) linéaire:

$$\min \sum_{i=1}^{n} c_i x_i$$

tel que

$$x_i \geq 0 \ \forall i \text{ et } \sum_{i=1}^{n} a_{ji} x_i = b_j \ \forall j$$

En notation matricielle

$$\min c^{\mathrm{T}} x \text{ tel que } Ax = b \text{ et } x \geq 0$$

Ce problème est parmi les plus simples, et peut être résolu à pour de grandes tailles (m et $n \approx 10^7$)

B. Problème d'affectation

Étant donné

- n personnes à occuper
- n tâches à accomplir
- la connaissance du temps qu'il faut à chacune des personnes pour exécuter chacune des tâches

Affecter (de manière bijective) les n tâches aux n personnes de manière à minimiser le temps total passé à effectuer ces tâches

Il s'agit d'un problème en variables discrètes qui comporte a priori un nombre exponentiel de solutions potentielles $(n!) \rightarrow$ énumération explicite impossible en pratique

Formulation

Première idée: x_i contient le numéro de la tâche de la personne i (n variables entières comprises entre 1 et n)

Problème : comment exprimer la bijectivité ?

Meilleure formulation:

- Indices i pour les personnes ($1 \leq i \leq n$)
- Indices j pour les tâches ($1 \leq j \leq n$)
- Données :

  $a_{ij} \longrightarrow$ durée de la tâche j pour la personne i
- Inconnues:

  x_{ij} variable binaire $\{0, 1\}$ indiquant si la personne i effectue la tâche j

Formulation (suite)

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

tel que

$$\sum_{i=1}^{n} x_{ij} = 1 \; \forall j, \; \sum_{j=1}^{n} x_{ij} = 1 \; \forall i, \quad \text{et } x_i \in \{0,1\} \; \forall i$$

- Nombre plus élevé de variables (n^2) $\longrightarrow$ plus difficile ?
- Problème linéaire en nombre entiers (variable binaires) $\longrightarrow$ méthodes de résolution différentes
- Mais formulation de la bijectivité simplifiée

Malgré le nombre exponentiel de solutions potentielles, ce problème peut être résolu très efficacement ! ($n \gg 10^6$)

C. Problème du voyageur de commerce

Étant donnés

- ◇ un voyageur de commerce désirant visiter n villes lors d'un circuit passant une et une seule fois parmi chacune de ces villes
- ◇ la connaissance de la distance (ou le temps de parcours) entre chaque paire de villes

Trouver le circuit optimal, visitant chaque ville une et une seule fois et de longueur (ou durée) minimale

Problème également de nature discrète et exponentielle

Autres application : soudure sur circuits intégrés

Formulation

Première idée: x_i contient le numéro de la i à visiter sur le circuit (n variables entières comprises entre 1 et n)

Problèmes : comment exprimer que chaque ville est visitée?

Meilleure formulation:

- Indices i et j pour les villes ($1 \leq i, j \leq n$)
- Données :

  $a_{ij} \longrightarrow$ distance (ou durée du trajet) entre i et j
- Inconnues:

  x_{ij} variable binaire $\{0, 1\}$ indiquant si le circuit passe de la ville i à la ville j

Formulation (suite)

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

tel que

$$\sum_{i=1}^{n} x_{ij} = 1 \; \forall j, \; \sum_{j=1}^{n} x_{ij} = 1 \; \forall i, x_i \in \{0, 1\} \; \forall i$$

et $\displaystyle\sum_{i \in S, j \notin S} x_{ij} \geq 1 \, \forall S$ avec $S \subseteq \{1, 2, \ldots, n\}$ et $1 < |S| < n$

- ◇ Nombre élevé (exponentiel) de contraintes
- ◇ Problème beaucoup plus difficile à résoudre ($n \approx 10^4$)

Algorithmes et complexité

Pourquoi ces trois problèmes sont-ils différents ?

Trois problèmes linéaires: a priori les plus simples ... ?

- ◇ A. Régime: variables continues → optimisation linéaire
- ◇ B. Affectation: variables discrètes, solutions en nombre exponentiel

  → optimisation linéaire en nombre entiers (mais ...)
- ◇ C. Voyageur: variables discrètes, solutions et contraintes en nombre exponentiel

  → optimisation linéaire en nombre entiers

Pourtant, B n'est pas plus difficile que A tandis que C est beaucoup plus difficile que A et B !

Complexité algorithmique

La difficulté d'un problème découle de l'efficacité des méthodes de résolution (algorithmes) qu'on peut lui appliquer

$\Rightarrow$ qu'est-ce qu'un bon algorithme ?

- ◇ Résout (approximativement) le problème
- ◇ Jusqu'au milieu du 20^{e} siècle: dans un temps fini
- ◇ A présent (ordinateurs): dans un temps (nombre d'opéra–tions élémentaires) borné (par la taille du problème)
  $\longrightarrow$ complexité algorithmique (de pire cas / moyenne)

Distinction essentielle:
complexité polynomiale $\leftrightarrow$ exponentielle

Méthodes de résolution pour l'optimisation linéaire

Pour l'optimisation linéaire en variables continues,
on dispose d'algorithmes très performants ($n \approx 10^7$)

- Algorithme du simplexe (1947)
  Complexité *exponentielle* mais ...
  Très performant en pratique
- Méthode de l'ellipsoïde (1978)
  Complexité *polynomiale* mais ...
  Peu performante en pratique
- Méthodes de point intérieur (1985)
  Complexité *polynomiale* et ...
  Très performantes en pratique (grands problèmes)

Méthodes de résolution pour l'optimisation linéaire (suite)

Pour l'optimisation linéaire en nombres entiers, les algorithmes sont beaucoup moins performants, car le problème est de nature intrinsèquement exponentielle
(cf. classe de problèmes *NP-complets*)

- Relaxation linéaire (approximation)
- Méthodes de séparation et évaluation

  Complexité *exponentielle*

$\longrightarrow$ Des problèmes de taille moyenne voire réduite ($n \approx 10^2$) peuvent déjà poser des difficultés insurmontables

$\longrightarrow$ On résout C. beaucoup moins efficacement que A.

Le cas particulier du problème d'affectation B.

Pourquoi peut-on résoudre efficacement ce problème ?

Parce qu'on peut le simplifier: on peut montrer que remplacer les variables $x_{ij} \in \{0, 1\}$ par $0 \le x_{ij} \le 1$ ne change pas la valeur optimale du problème !

On obtient un problème d'optimisation linéaire en variables continues $\longrightarrow$ Importance de la reformulation

En général, si on peut remplacer les variables binaires par des variables continues et un nombre polynomial de contraintes linéaires additionnelles, on peut résoudre le problème résultant en temps polynomial

$\longrightarrow$ tous les problèmes combinatoires/en nombres entiers ne sont pas difficiles !

Dualité

Forme canonique

Soit le problème linéaire (en les m variables y_i)

$$\max \sum_{i=1}^{m} b_i y_i \text{ tel que } \sum_{i=1}^{m} a_{ij} y_i \leq c_j \ \forall 1 \leq j \leq n$$

(objectif et n contraintes d'inégalités linéaire), ou encore

$$\max b^{\mathrm{T}} y \text{ tel que } A^{\mathrm{T}} y \leq c$$

(forme matricielle avec $b, y \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ et $A \in \mathbb{R}^{m \times n}$)

Tous les problèmes linéaires peuvent être exprimés sous cette forme (après reformulation équivalente)

Bornes et optimalité

Soit une solution admissible $\bar{y}$ (vérifiant la contrainte $A^T y \leq c$) $\rightarrow b^T \bar{y}$ est une borne inférieure sur la valeur optimale du problème p^*

Mais comment

- obtenir une borne supérieure sur cette valeur optimale
- prouver qu'une solution donnée y^* est optimale

Ces deux questions sont liées car

$$\text{prouver que } y^* \text{ est optimale}$$
$$\Updownarrow$$
$$\text{prouver que } b^T y^* \text{ est une borne supérieure sur la valeur de l'optimum } p^*$$

Générer des bornes supérieures

Soit le petit problème

$$\max y_1 + 2y_2 + 3y_3 \text{ tel que } \begin{array}{rcll} y_1 + y_2 & \leq & 1 & (a) \\ y_2 + y_3 & \leq & 2 & (b) \\ y_3 & \leq & 3 & (c) \end{array}$$

La solution $x = (1, 0, 2)$ est admissible et correspond à un objectif de valeur 7 $\rightarrow$ borne inférieure $p^* \geq 7$

Combinons les contraintes selon $(a) + (b) + 2(c)$

$$y_1 + y_2 + y_2 + y_3 + 2y_3 \leq 1 + 2 + 2 \times 3 \Leftrightarrow y_1 + 2y_2 + 3y_3 \leq 9$$

$\rightarrow$ borne supérieure sur la valeur optimale $p^* \leq 9$

Comme de plus la solution admissible $x = (2, -1, 3)$ donne un objectif égal à 9, nous avons une preuve que $(2, -1, 3)$ est une solution optimale $\rightarrow p^* = 9$

La meilleure borne supérieure

Cherchons la meilleure borne supérieure que l'on peut obtenir par ce procédé

$$\max \sum_{i=1}^{m} b_i y_i \text{ tel que } \sum_{i=1}^{m} a_{ij} y_i \leq c_j \; \forall 1 \leq j \leq n$$

Considérons n variables x_i en guise de multiplicateurs pour les n contraintes, avec $x_i \geq 0$ (préserver le sens des inégalités) et additionnons les inégalités résultantes

$$\sum_{j=1}^{n} x_j \sum_{i=1}^{m} a_{ij} y_i \leq \sum_{j=1}^{n} x_j c_j \Leftrightarrow \sum_{i=1}^{m} y_i (\sum_{j=1}^{n} a_{ij} x_j) \leq \sum_{j=1}^{n} c_j x_j$$

La meilleure borne supérieure (suite)

Cela donne une borne supérieure $\sum_{j=1}^n c_j x_j$ sur l'objectif

$$\text{pour autant que } \sum_{j=1}^n a_{ij} x_j = b_i \; \forall 1 \leq i \leq m$$

Minimisons à présent cette borne supérieure

$$\min \sum_{j=1}^n c_j x_j \text{ tel que } \sum_{j=1}^n a_{ij} x_j = b_i \; \forall 1 \leq i \leq m \text{ et } x_i \geq 0$$

ou encore

$$\min c^{\mathrm{T}} x \text{ tel que } Ax = b \text{ et } x \geq 0$$

C'est un autre problème d'optimisation linéaire appelé problème dual du problème d'origine (primal) !

Propriétés de dualité

- Toute solution admissible pour le dual (resp. primal) fournit une borne supérieure (resp. inférieure) pour le primal (resp. dual)

  (conséquence de la dérivation du problème dual)

- L'inégalité $b^{\mathrm{T}} y \leq c^{\mathrm{T}} x$ est valable pour tout x, y tels que $Ax = b$, $x \geq 0$ et $A^{\mathrm{T}} y \leq c$ (corollaire)

- La valeur optimale du problème dual d^* est toujours égale à la valeur optimale du problème primal

Cette dernière propriété est non-triviale et se nomme propriété du dualité forte $\longrightarrow$ il est ainsi toujours possible de trouver une *preuve* qu'une solution est optimale !

Autres propriétés et conséquences

- On peut vérifier que le dual du problème dual est équivalent au problème primal d'origine
- On peut indifféremment résoudre le problème primal ou le problème dual pour trouver la valeur optimale de l'objectif
- Certains algorithmes de type primal-dual résolvent les deux problèmes simultanément
- La notion de problème dual peut se généraliser dans le cas de l'optimisation non-linéaire mais la propriété du dualité forte peut cesser d'être valide

Optimisation non-linéaire

L'optimisation linéaire ne permet pas de modéliser toutes les situations de façon satisfaisante $\longrightarrow$ revenons à

$$\min_{x \in \mathbb{R}^n} f(x) \text{ tel que } x \in \mathcal{D} \subseteq \mathbb{R}^n$$

Retour à la complexité

On a vu qu'un ensemble $\mathcal{D}$ discret peut rendre le problème difficile (complexité exponentiel) mais ce problème peut également être difficile dans le cas continu
Par exemple, le problème *simple* $\min f(x_1, x_2, \ldots, x_{10})$ peut nécessiter jusqu'à 10^{20} opérations pour garantir l'obtention d'une solution avec précision de 1% !

Deux approches distinctes

◇ S'attaquer à tous les problèmes sans garantie d'efficacité

– Optimisation non-linéaire classique

– Méthodes (méta)-heuristiques

◇ Se limiter à certains type de problème avec en contrepartie une garantie d'efficacité

– Optimisation linéaire

  * algorithmes spécialisés très rapides
  * mais trop limitée en pratique

– Optimisation **convexe**

Compromis généralité $\leftrightarrow$ efficacité

Qu'est-ce que l'optimisation convexe ?

- Domaine admissible convexe tel que $\forall x, y \in \mathcal{D}$

  le segment joignant x et y appartient entièrement à $\mathcal{D}$

- Fonction objectif convexe

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y) \; \forall x, y$$

- Cas particulier: programmation linéaire
- Cas particulier: programmation quadratique avec des formes quadratiques semidéfinies positives
- De nombreux autres problèmes sont convexes (ou convexifiables)

Avantage: algorithmes *efficaces* (polynomiaux)

Optimisation conique convexe

Objectif: généraliser la programmation linéaire

$$\max b^T y \text{ tel que } A^T y \leq c$$

$$\min c^T x \text{ tel que } Ax = b \text{ et } x \geq 0$$

en gardant ses bonnes propriétés

- dualité
- algorithmes efficaces

Idée: généraliser les signes d'inégalité $\leq$ et $\geq$

Quelles sont les *bonnes* propriétés de ces inégalités ?

Généraliser $\geq$ et $\leq$

Soit un ensemble $K \subseteq \mathbb{R}^n$. Définissons

$$a \succeq_K 0 \Leftrightarrow a \in K$$

On a aussi

$$a \succeq_K b \Leftrightarrow a - b \succeq_K 0 \Leftrightarrow a - b \in K$$

de même que

$$a \preceq_K b \Leftrightarrow b \succeq_K a \Leftrightarrow b - a \succeq_K 0 \Leftrightarrow b - a \in K$$

Imposons deux propriétés raisonnables

$$a \succeq_K 0 \Rightarrow \lambda a \succeq_K 0 \ \forall \lambda \geq 0$$

ainsi que

$$a \succeq_K 0 \text{ et } b \succeq_K 0 \Rightarrow a + b \succeq_K 0$$

Optimisation conique

On peut alors généraliser

$$\max b^{\mathrm{T}} y \text{ tel que } A^{\mathrm{T}} y \leq c$$

en

$$\max b^{\mathrm{T}} y \text{ tel que } A^{\mathrm{T}} y \preceq_K c$$

(le cas linéaire correspond à $K = \mathbb{R}^n_+$)

Propriétés

- ◇ Ce problème est convexe

  (en fait, K est un cône convexe car fermé pour l'addition et la multiplication par un scalaire positif)

- ◇ Tous les problèmes convexes peuvent être exprimés sous forme conique

Propriétés de dualité

Si on effectue la même démarche pour obtenir des bornes supérieures, on obtient à la place du problème dual linéaire

$$\min c^{\mathrm{T}} x \text{ tel que } Ax = b \text{ et } x \geq 0$$

le problème dual suivant

$$\min c^{\mathrm{T}} x \text{ tel que } Ax = b \text{ et } x \succeq_{K^*} 0$$

où l'ensemble K^* est défini par

$$K^* = \{z \in \mathbb{R}^n \text{ tel que } x^{\mathrm{T}} z \geq 0 \ \forall x \in K\}$$

de façon à garder la propriété permettant l'aggrégation de n inégalités en une borne supérieure

L'ensemble K^* est également un cône convexe, on l'appelle le cône dual de K ($K^* = \mathbb{R}^n_+$ dans le cas linéaire)

Propriétés de la paire primale-duale

$$\max b^{\mathrm{T}} y \text{ tel que } A^{\mathrm{T}} y \preceq_K c$$
$$\min c^{\mathrm{T}} x \text{ tel que } Ax = b \text{ et } x \succeq_{K^*} 0$$

- ◇ Formulation très symétrique, calcul de K^* seule difficulté pour calculer le dual
- ◇ Nonlinéarité confinée à l'intérieur du cône
  $\longrightarrow$ avantage théorique et pratique
- ◇ Dualité faible: $b^{\mathrm{T}} y \leq c^{\mathrm{T}} x$ pour tout x, y admissibles ($Ax = b$, $x \succeq_{K^*} 0$ et $A^{\mathrm{T}} y \preceq_K c$)
- ◇ Dualité forte: sous une certaine condition (peu restrictive), on a l'égalité $b^{\mathrm{T}} y^* = c^{\mathrm{T}} x^*$

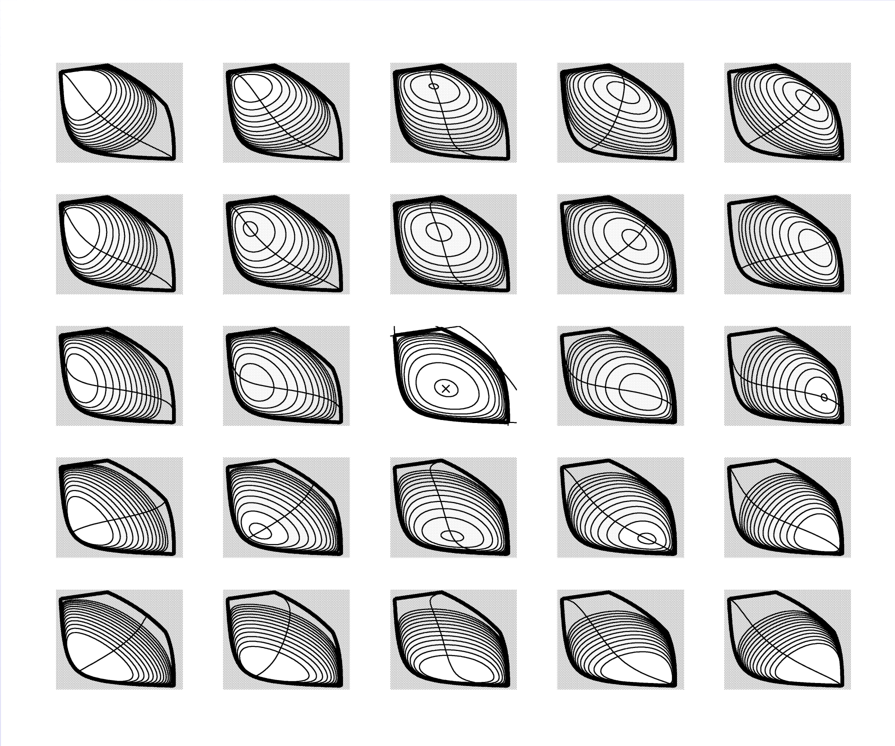
- ◇ Généralisation possible des méthodes de point intérieur pour l'optimisation linéaire à l'optimisation conique convexe $\longrightarrow$ algorithmes efficaces

Exemple. L'optimisation semidéfinie correspond au cas $K = \mathbb{S}^n_+$ (matrices symétriques semidéfinies positives)
On a dans ce cas

- ◇ $K^* = \mathbb{S}^n_+ = K$: le dual d'un problèmes d'optimisation semidéfinie est un problème d'optimisation semidéfinie
- ◇ Applications dans de nombreux domaines
  - contrôle
  - statistiques
  - relaxations de problèmes combinatoires difficiles

Conclusions

- L'optimisation permet de résoudre de nombreux problèmes d'aide à la décision dans de multiples domaines
- L'optimisation linéaire est un cas particulier particulièrement bien étudié (en théorie et en pratique)
- La théorie de dualité révèle l'existence d'un problème dual fortement lié au problème primal
- L'optimisation convexe est une généralisation naturelle de l'optimisation linéaire (sous forme conique)
- Il est possible de résoudre efficacement les problèmes linéaires et convexes



Merci de votre attention