R5.A.11 Méthodes d'optimisation

Thibault Godin
IUT de Vannes Informatique

R5.A.11 Méthodes d'optimisation

Déroulé :

- 45min de CM 1 semaine sur 2
- 1h30 de TP toutes les semaines

Évaluation :

- QCM en fin de CM
- 1 CT
- pour les FI 1 mini projet/rapport à soutenir (binôme)

R5.A.11 Méthodes d'optimisation

Déroulé :

- 45min de CM 1 semaine sur 2
- 1h30 de TP toutes les semaines

Évaluation :

- QCM en fin de CM
- 1 CT
- pour les FI 1 mini projet/rapport à soutenir (binôme)

Programme :

- Programmation linéaire *simplexe, système d'équation, flots*
- Recuit simulé *minimisation, voyageur de commerce, sudoku*
- Clustering *classement, prédiction*

IAg

Par défaut, le module est en IAg niveau 2 **utilisation limitée** l'utilisation est autorisée pour améliorer marginalement un travail produit par la personne étudiante.

IAg

Par défaut, le module est en IAg niveau 2 **utilisation limitée** l'utilisation est autorisée pour améliorer marginalement un travail produit par la personne étudiante.

Usages possibles :

- Analyser des contenus
- Obtenir une rétroaction
- Évaluer la qualité de son travail à partir de critères
- Demander à être confronté relativement à ses idées, à sa démarche
- Diriger les processus de résolution de problèmes

IAg

Par défaut, le module est en IAg niveau 2 **utilisation limitée** l'utilisation est autorisée pour améliorer marginalement un travail produit par la personne étudiante.

Usages possibles :

- Analyser des contenus
- Obtenir une rétroaction
- Évaluer la qualité de son travail à partir de critères
- Demander à être confronté relativement à ses idées, à sa démarche
- Diriger les processus de résolution de problèmes

Cela vient bien entendu en complément des enseignants qui restent votre ressource à privilégier.

Certaines parties (en particulier en TP) pourront être de niveau IAg supérieur (3). Cela sera précisé.

Les évaluations toutes papier seront niveau 0 : IA interdite. Le projet sera niveau 3 et une trace et documentation de l'usage éventuel d'une IAg

Optimisation : recherche de minimum

Beaucoup de problèmes d'optimisation se placent dans la catégorie "minimiser/maximiser une fonction" (souvent un coût/profit ou une énergie)

Optimisation : recherche de minimum

Beaucoup de problèmes d'optimisation se placent dans la catégorie "minimiser/maximiser une fonction" (souvent un coût/profit ou une énergie)

Exemple :

- Minimiser le temps passer à réviser les maths pour avoir 12/20
- Minimiser la distance parcourue en voiture
- Problème d'emploi du temps : on fait une fonction de coût/énergie qui pénalise fortement les chevauchement et faiblement les contraintes
- Optimiser son deck d'hearthstone

Optimisation : recherche de minimum

Beaucoup de problèmes d'optimisation se placent dans la catégorie "minimiser/maximiser une fonction" (souvent un coût/profit ou une énergie)

Exemple :

- Minimiser le temps passer à réviser les maths pour avoir 12/20
- Minimiser la distance parcourue en voiture
- Problème d'emploi du temps : on fait une fonction de coût/énergie qui pénalise fortement les chevauchement et faiblement les contraintes
- Optimiser son deck d'hearthstone

On passe généralement de la mini$^{\text{misation}}$ à la maxi$^{\text{misation}}$ en prenant l'opposé ou l'inverse de la fonction objectif

Avant des commencer, une précaution qui risque d'en décevoir certains : la **programmation linéaire** ne parle pas vraiment de programmation. Le terme tend d'ailleurs à être remplacer par *optimisation linéaire* (sous contraintes)

Comme très souvent, une excellente source est le **Cormen, Leiserson, Rivest** *Introduction à l'algorithmique* (disponible à la BU)

Vous pouvez aussi consulter https://www2.mat.ulaval.ca/fileadmin/Cours/MAT-2920/Chapitre3.pdf qui donne de bon exemples.

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

- Comment enseigner un maximum de concepts mathématiques

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

- Comment enseigner un maximum de concepts mathématiques *en moins de 3h*
- Comment avoir toutes ses compétences en BUT1

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

- Comment enseigner un maximum de concepts mathématiques *en moins de 3h*
- Comment avoir toutes ses compétences en BUT1 *en ayant moins de 6 en maths*
- Comment maximiser les calories de ses courses

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

- Comment enseigner un maximum de concepts mathématiques *en moins de 3h*
- Comment avoir toutes ses compétences en BUT1 *en ayant moins de 6 en maths*
- Comment maximiser les calories de ses courses *en dépensant moins de 100 €*

Optimisation sous contraintes (linéaires)

Dans beaucoup de cas en optimisation on veut maximiser une quantité, mais on a des contraintes qui nous limitent :

- Comment enseigner un maximum de concepts mathématiques *en moins de 3h*
- Comment avoir toutes ses compétences en BUT1 *en ayant moins de 6 en maths*
- Comment maximiser les calories de ses courses *en dépensant moins de 100 €*

On va se limiter à des contraintes et des objectifs linéaires

Système d'inéquations

On veut crafter des objets dans mIUTcraft™® ©

Un ordinateur nécessite 1 verre, 1 plastique et 9 silicium ; une tablette utilise 2 verres, 1 plastique et 4 silicium

On a seulement 8 verres, 5 plastiques et 36 silicium

Un ordinateur se vend 2 BUTin et une tablette 1 BUTin. On cherche à maximiser la quantité de BUTin totale.

Système d'inéquations

On veut crafter des objets dans mIUTcraft™® ©

Un ordinateur nécessite 1 verre, 1 plastique et 9 silicium ; une tablette utilise 2 verres, 1 plastique et 4 silicium

On a seulement 8 verres, 5 plastiques et 36 silicium

Un ordinateur se vend 2 BUTin et une tablette 1 BUTin. On cherche à maximiser la quantité de BUTin totale.

On obtient le problème :

$$\text{Maximiser} \quad c(x, y) = 2x + y \quad \text{sous les contraintes} \quad \begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \end{cases}$$

Système d'inéquations

On veut crafter des objets dans mIUTcraft™® ©

Un ordinateur nécessite 1 verre, 1 plastique et 9 silicium ; une tablette utilise 2 verres, 1 plastique et 4 silicium

On a seulement 8 verres, 5 plastiques et 36 silicium

Un ordinateur se vend 2 BUTin et une tablette 1 BUTin. On cherche à maximiser la quantité de BUTin totale.

On obtient le problème :

Maximiser $\quad c(x,y) = 2x + y \quad$ sous les contraintes $\quad \begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \end{cases}$

Voc. : on a 2 variables (x, y) et 3 équations. C'est donc un problème 2d
rmq. : contrainte sous-entendue au problème : $x, y \geq 0$
rmq : on va commencer par mettre de côté la restriction "les nombres sont entiers"

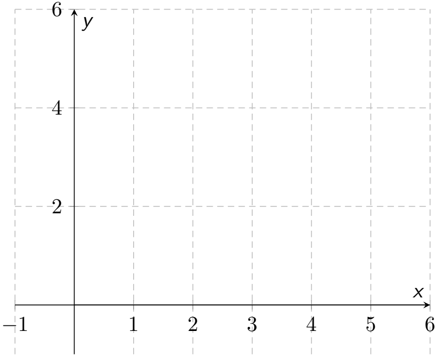
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



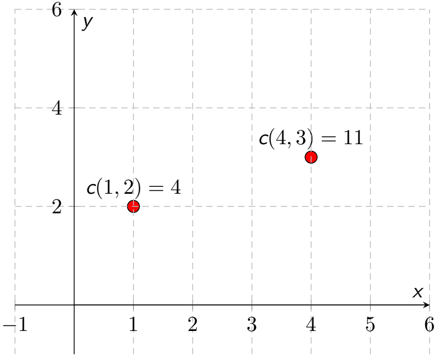
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



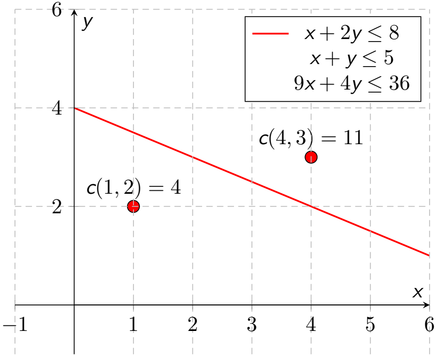
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



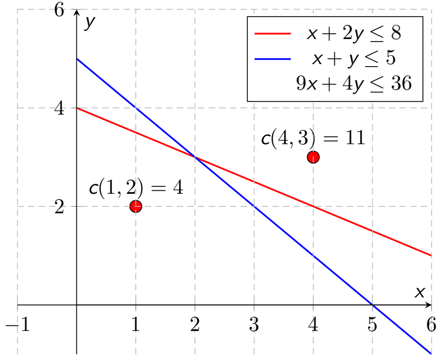
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



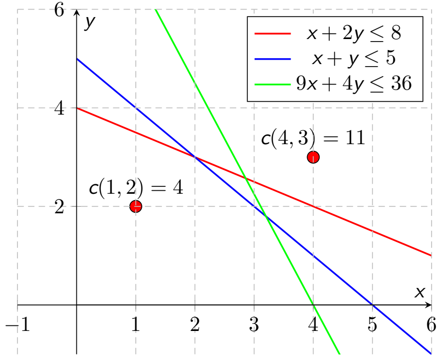
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



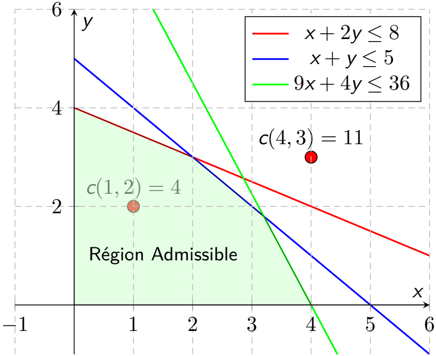
Représentation graphique du problème

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Méthodes de résolution

- Graphiques
  - Par isoclines (courbes de niveau)
  - Par les points extrémaux (sommets)
- Algorithmiques
  - Méthode du simplexe (variation du pivot de Gauss)
  - Méthodes polynomiales (ellipsoïde, Karmarkar ...)

Résolution graphique : gradient et lignes de niveau

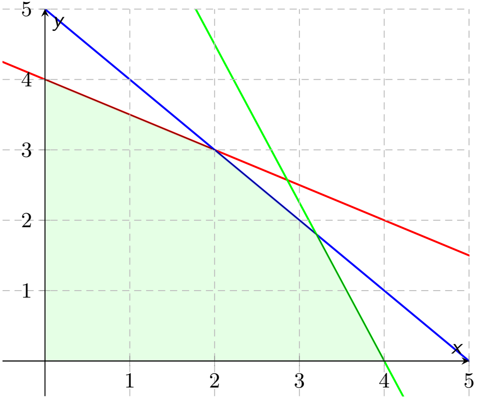
La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

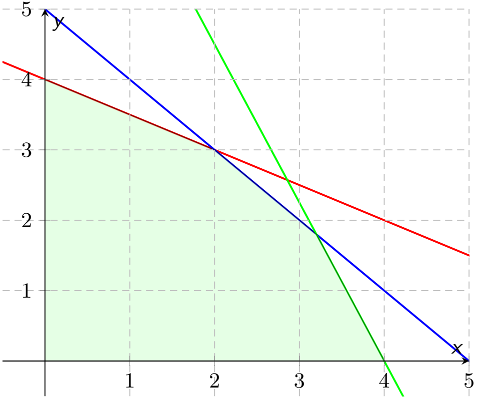
La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$
forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

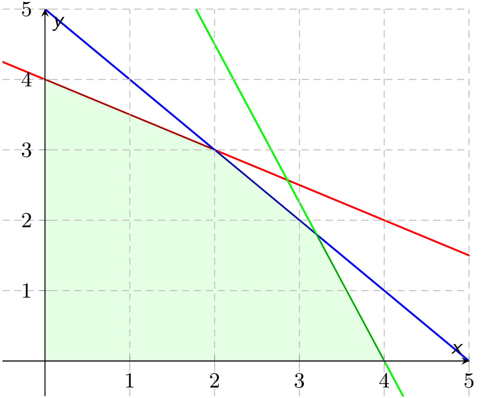
les droites de niveau sont perpendiculaires au gradient

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

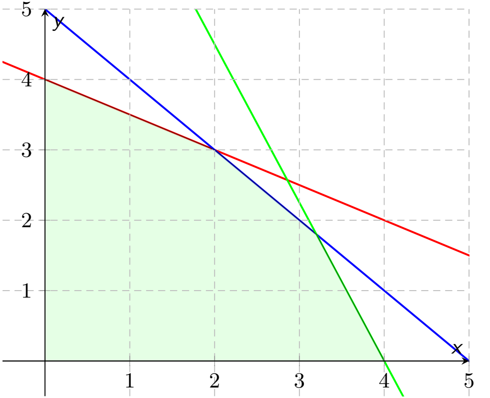
en restant dans la région admissible

Maximiser

$c(x, y) = 2x + y$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

en restant dans la région admissible

$$\nabla c(x, y) = (2, 1)$$

$$c(x, y) = cst \iff 2x + y = cst$$

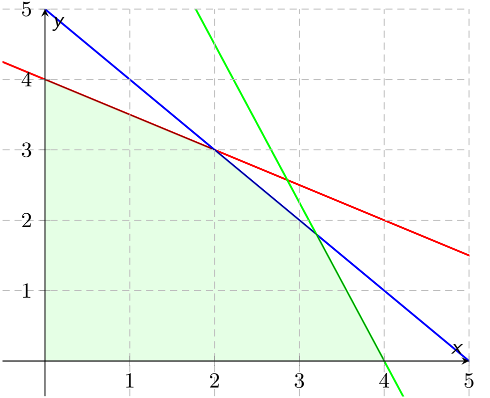
$$\iff y = cst - 2x$$

Maximiser

$c(x, y) = 2x + y$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

en restant dans la région admissible

$$\nabla c(x, y) = (2, 1)$$

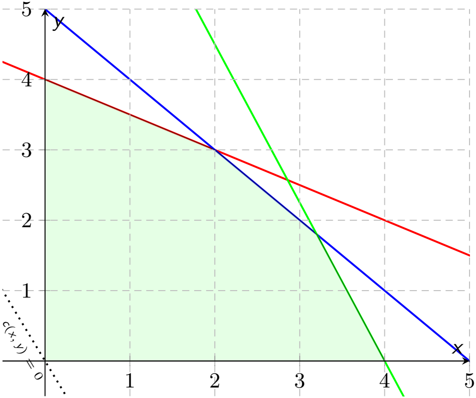
$$c(x, y) = cst \iff 2x + y = cst$$
$$\iff y = cst - 2x$$

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x, y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

en restant dans la région admissible

$$\nabla c(x, y) = (2, 1)$$

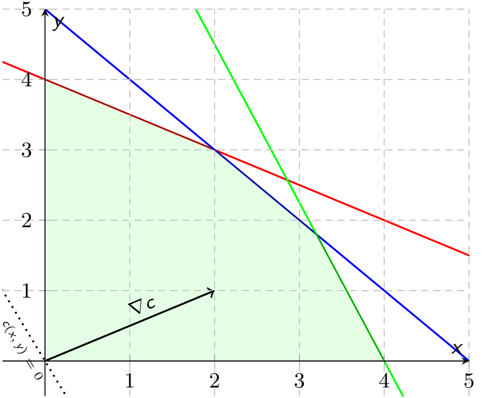
$$\begin{aligned} c(x, y) = cst &\iff 2x + y = cst \\ &\iff y = cst - 2x \end{aligned}$$

Maximiser

$$c(x, y) = 2x + y$$

sous les contraintes

$$\begin{cases} x \quad +2y \quad \leq 8 \\ x \quad +y \quad \leq 5 \\ 9x \quad +4y \quad \leq 36 \\ \quad x, y \quad \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x,y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

en restant dans la région admissible

$$\nabla c(x,y) = (2,1)$$

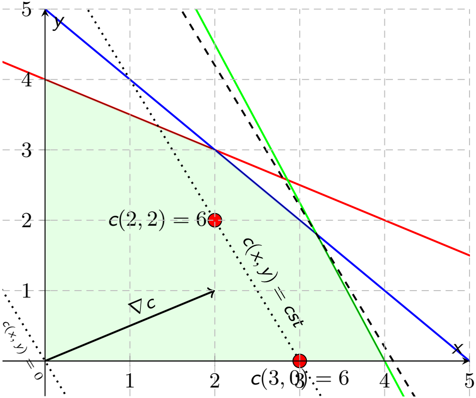
$$c(x,y) = cst \iff 2x + y = cst$$
$$\iff y = cst - 2x$$

Maximiser

$$c(x,y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x,y & \geq 0 \end{cases}$$



Résolution graphique : gradient et lignes de niveau

La fonction de coût $c(x,y)$ est une fonction (forme linéaire) continue $\rightsquigarrow$

forme linéaire $\rightsquigarrow$ les courbes de niveau sont des droites (en 2d, plan en 3d ...)

calcul du gradient $\rightsquigarrow$ direction de recherche (optimisation classique)

les droites de niveau sont perpendiculaires au gradient

en restant dans la région admissible

$$\nabla c(x,y) = (2,1)$$

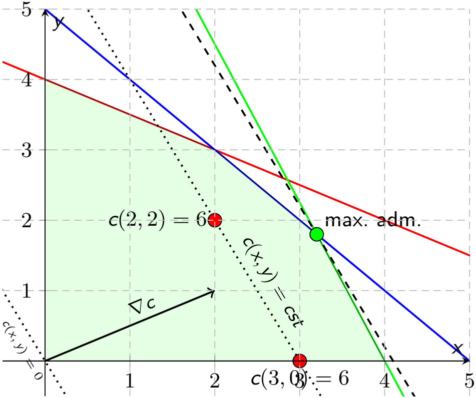
$$\begin{aligned} c(x,y) = cst &\iff 2x + y = cst \\ &\iff y = cst - 2x \end{aligned}$$

Maximiser

$c(x,y) = 2x + y$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x,y & \geq 0 \end{cases}$$



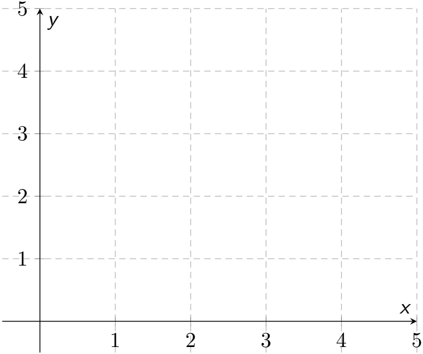
Résolution graphique : résumé

Maximiser

$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



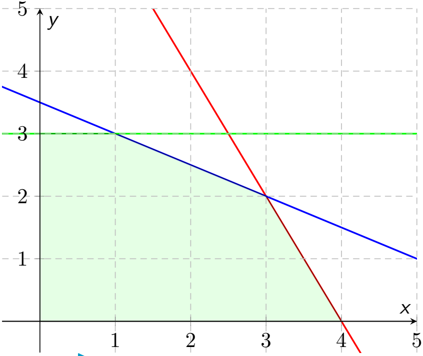
Résolution graphique : résumé

Maximiser

$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



- tracer la région admissible
-

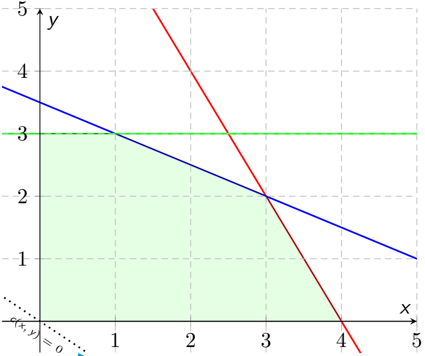
Résolution graphique : résumé

Maximiser

$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



- tracer la région admissible
- calculer la droite $c(x, y) = 0$

-
- $y = -\frac{4}{5}x$

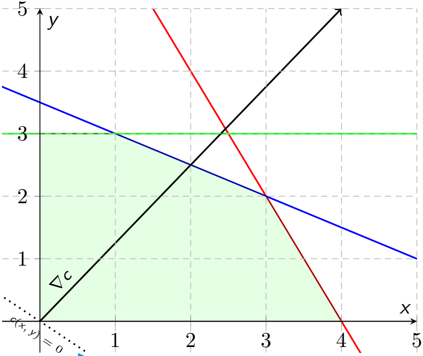
Résolution graphique : résumé

Maximiser

$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



- tracer la région admissible
- calculer la droite $c(x,y) = 0$
- calculer $\nabla c(x,y)$

-
- $y = -\frac{4}{5}x$
- $\nabla c = (4,5)$

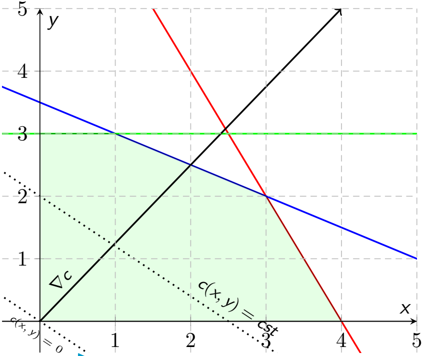
Résolution graphique : résumé

Maximiser

$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



- tracer la région admissible
- calculer la droite $c(x, y) = 0$
- calculer $\nabla c(x, y)$
- chercher la parallèle la plus éloignée dans la direction de ∇c qui reste dans la zone admissible

-
- $y = -\frac{4}{5}x$
- $\nabla c = (4, 5)$

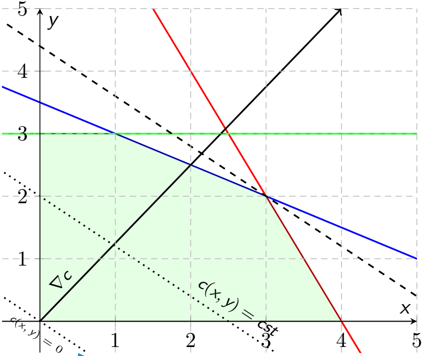
Résolution graphique : résumé

Maximiser

$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x,y & \geq 0 \end{cases}$$



- tracer la région admissible
- calculer la droite $c(x,y) = 0$
- calculer $\nabla c(x,y)$
- chercher la parallèle la plus éloignée dans la direction de ∇c qui reste dans la zone admissible

-
- $y = -\frac{4}{5}x$
- $\nabla c = (4,5)$

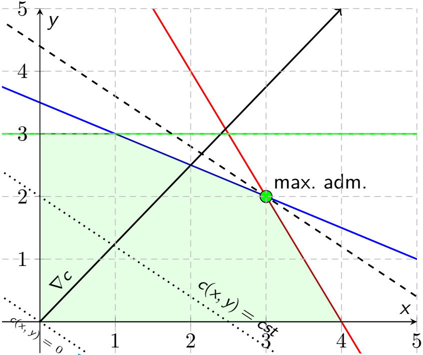
Résolution graphique : résumé

Maximiser

$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$



- tracer la région admissible
- calculer la droite $c(x, y) = 0$
- calculer $\nabla c(x, y)$
- chercher la parallèle la plus éloignée dans la direction de ∇c qui reste dans la zone admissible

-
- $y = -\frac{4}{5}x$
- $\nabla c = (4, 5)$
- le max est atteint en $(3, 2)$

Optimisation linéaire : écueils (I)

Minimiser

$c(x, y) = 4x + 5y$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

$$\downarrow$$

Maximiser

$c'(x, y) = -(4x + 5y)$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Minimiser

$c(x, y) = 4x + 5y$

sous les contraintes

$$\begin{cases} 2x & +y & \geq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

$$\downarrow$$

Maximiser

$c(x, y) = 4x + 5y$

sous les contraintes

$$\begin{cases} -2x & +-y & \leq -8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

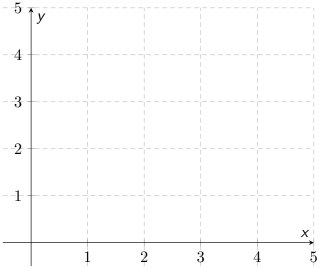
Résolution graphique : écueils (II)

Maximiser

$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x,y & \geq 0 \end{cases}$$



Résolution graphique : écueils (II)

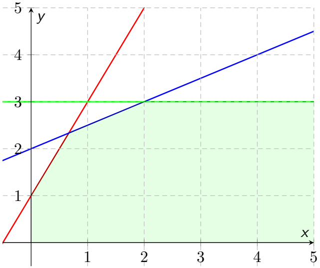
Maximiser

$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Région admissible non-bornée



Résolution graphique : écueils (II)

Maximiser

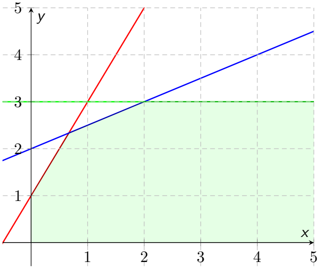
$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Région admissible non-bornée

pas de solution$/\max = +\infty$



Résolution graphique : écueils (II)

Maximiser

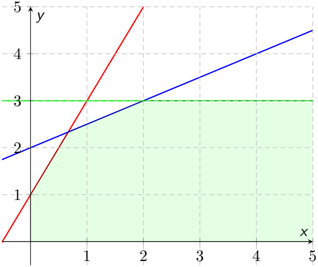
$$c(x,y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Région admissible non-bornée

pas de solution/$\max = +\infty$



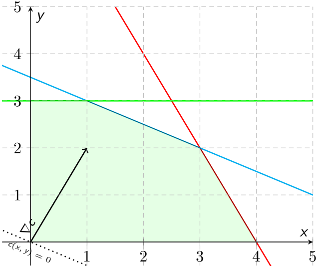
Maximiser

$$c(x,y) = x + 2y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

maximum unique



Résolution graphique : écueils (II)

Maximiser

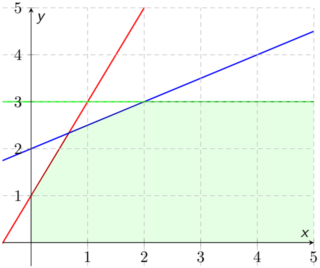
$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Région admissible non-bornée

pas de solution$/\max = +\infty$



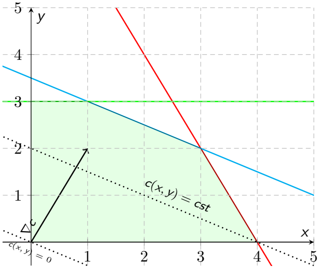
Maximiser

$$c(x, y) = x + 2y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

maximum unique



Résolution graphique : écueils (II)

Maximiser

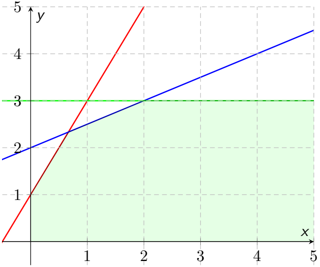
$$c(x, y) = 4x + 5y$$

sous les contraintes

$$\begin{cases} -2x & +y & \leq 1 \\ x & -2y & \leq -4 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

Région admissible non-bornée

pas de solution/$\max = +\infty$



Maximiser

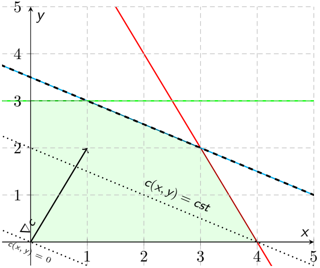
$$c(x, y) = x + 2y$$

sous les contraintes

$$\begin{cases} 2x & +y & \leq 8 \\ x & +2y & \leq 7 \\ & y & \leq 3 \\ & x, y & \geq 0 \end{cases}$$

infinité de solutions (x, y)

maximum unique



Résolution graphique par isoclines bilan

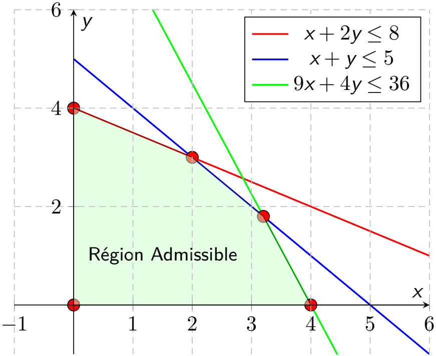
- $+$ Facile à la main (petits exemples)
- $-$ Peu précis à la main
- $-$ Passe mal à l'échelle (si bcp de contraintes)
- $-$ Passe mal à l'échelle (si bcp de variable $\rightsquigarrow$ 3D, 4D, ... nD)
- $-$ Peu précis algorithmiquement

Au final, on se retrouve à examiner un nouveau PL avec la contrainte $c(x, y) \geq cst$ et à vérifier que la région admissible n'est pas vide $\rightsquigarrow$ mauvais algorithme

Sommets

Théorème

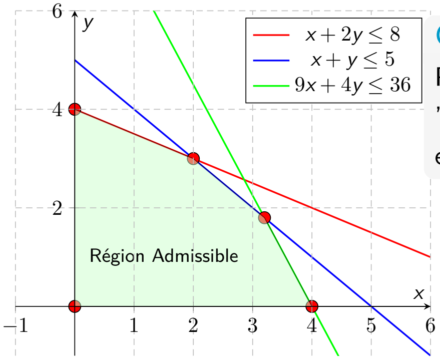
Si le polyèdre formé par l'ensemble des solutions d'un PL est borné, alors il existe au moins une solution optimale et l'une d'elles est obtenue sur un point extrême



Sommets

Théorème

Si le polyèdre formé par l'ensemble des solutions d'un PL est borné, alors il existe au moins une solution optimale et l'une d'elles est obtenue sur un point extrême

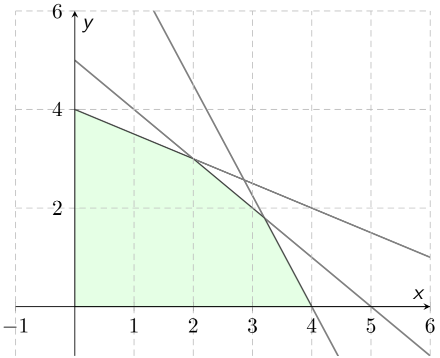


Corollaire

Pour trouver l'optimum, il "suffit" d'examiner les points extrêmes de la région admissible

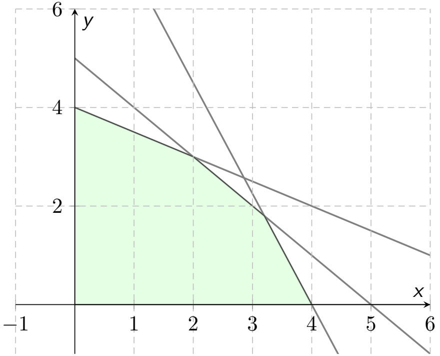
Résolution Graphique : Sommets

1. Partir d'un sommet ***s*** de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, ***s*** est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet ***t*** suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



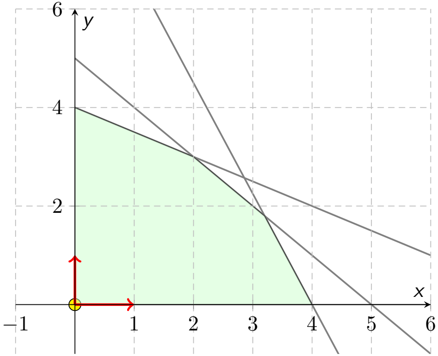
Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



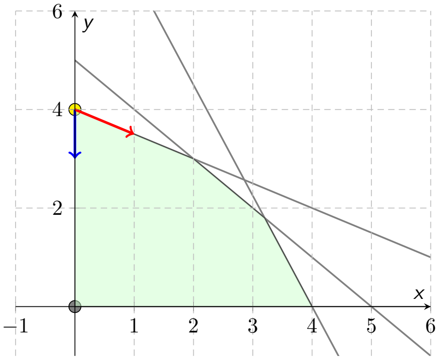
Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



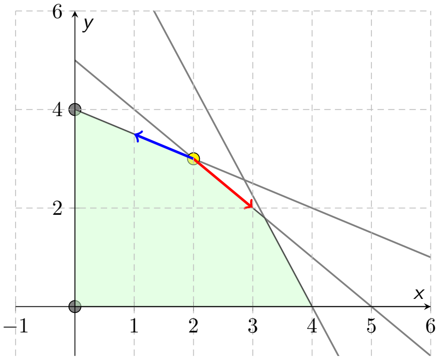
Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



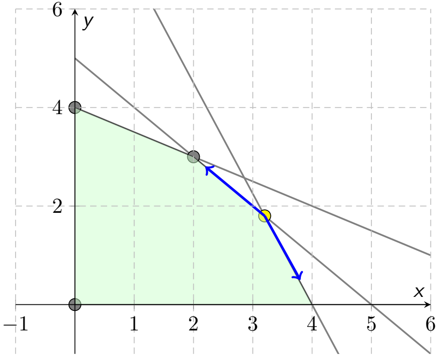
Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



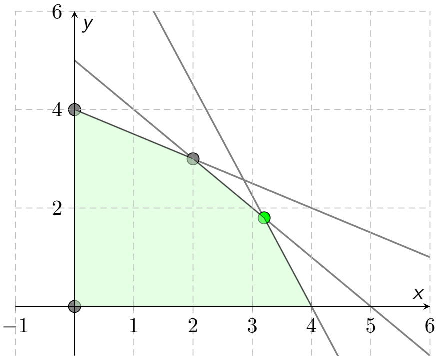
Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



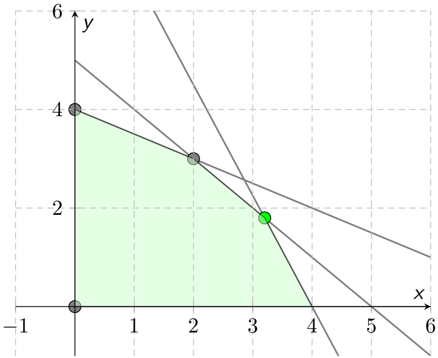
Résolution Graphique : Sommets

1. Partir d'un sommet ***s*** de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, ***s*** est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet ***t*** suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2



Résolution Graphique : Sommets

1. Partir d'un sommet $\boldsymbol{s}$ de la région admissible
2. Déterminer une arête le long de laquelle le coût augmente. S'il n'en existe pas, $\boldsymbol{s}$ est optimal, STOP
3. Se déplacer le long de l'arête jusqu'au sommet $\boldsymbol{t}$ suivant. S'il n'existe pas, le problème est non borné, STOP Sinon, poser $\boldsymbol{s} \leftarrow \boldsymbol{t}$ et revenir en 2

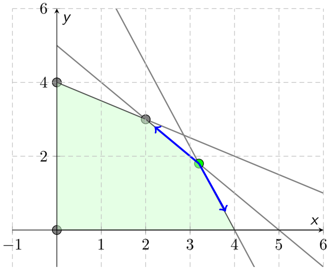


En 2D $\rightsquigarrow$ au pire nb contraintes sommets $\rightsquigarrow$ assez efficace

En 2D $\rightsquigarrow$ trouver un nouveau sommet $\rightsquigarrow$ assez efficace

Sommets : dimensions supérieures

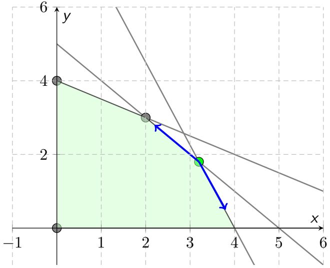
- en dimension supérieur il y a (beaucoup) de chemins/choix possibles
- les sommets sont plus difficiles à calculer en nD
- il peut y avoir jusqu'à $\Theta(c^{\lfloor \frac{n}{2} \rfloor})$ sommets pour un PL à n variables et c contraintes



Sommets : dimensions supérieures

- en dimension supérieur il y a (beaucoup) de chemins/choix possibles
- les sommets sont plus difficiles à calculer en nD
- il peut y avoir jusqu'à $\Theta(c^{\lfloor \frac{n}{2} \rfloor})$ sommets pour un PL à n variables et c contraintes

$\rightsquigarrow$ pas efficace algorithmiquement



Simplexe

Algorithme du simplexe $\rightsquigarrow$ Dantzig 1947

idée de base : raffinement de la méthode des sommets utilisant le pivot de Gauss (BUT1)

En écriture matricielle, un problème linéaire (de dimension q) sous forme normale s'écrit :

$$\text{Maximiser} \quad c^T x \quad \text{sous les contraintes} \quad \begin{cases} Ax & \leq b \\ x & \geq 0 \end{cases}$$

où

- $c = (c_1, ..., c_q)^T$ est le vecteur colonne (à q lignes) de calcul du coût ;
- $x = (x_1, ..., x_q)^T$ est le vecteurs colonnes (à q lignes) des variables ;
- $A = (a_{i,j})_{1 \leq i \leq p, 1 \leq j \leq q}$ est une matrice (à p lignes et q colonnes) des contraintes ;
- et $b = (b_1, ..., b_p)^T$ est le vecteur colonne (à p lignes) des bornes

Simplexe

Algorithme du simplexe $\rightsquigarrow$ Dantzig 1947

idée de base : raffinement de la méthode des sommets utilisant le pivot de Gauss (BUT1)

En écriture matricielle, un problème linéaire (de dimension q) sous forme normale s'écrit :

$$\text{Maximiser} \quad c^T x \quad \text{sous les contraintes} \quad \begin{cases} Ax & \leq b \\ x & \geq 0 \end{cases}$$

où

- $c = (c_1, ..., c_q)^T$ est le vecteur colonne (à q lignes) de calcul du coût ;
- $x = (x_1, ..., x_q)^T$ est le vecteurs colonnes (à q lignes) des variables ;
- $A = (a_{i,j})_{1 \leq i \leq p, 1 \leq j \leq q}$ est une matrice (à p lignes et q colonnes) des contraintes ;
- et $b = (b_1, ..., b_p)^T$ est le vecteur colonne (à p lignes) des bornes

Simplexe : forme matricielle (standard)

Maximiser

$$c(x,y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x, y & \geq 0 \end{cases}$$

$$A = \begin{pmatrix} 1 & 2 \\ 1 & 1 \\ 9 & 4 \end{pmatrix} \; b = \begin{pmatrix} 8 \\ 5 \\ 36 \end{pmatrix}$$

$$c = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \; X = \begin{pmatrix} x \\ y \end{pmatrix}$$

Maximiser $\quad c^T X$

sous les contraintes

$$\begin{cases} AX & \leq b \\ X & \geq 0 \end{cases}$$

Simplexe : forme matricielle standard

En fait, on cherche souvent une forme encore plus rigide, la *forme standard*

$$\text{Maximiser} \quad c^T x \quad \text{sous les contraintes} \quad \begin{cases} Ax & = b \\ x & \geq 0 \end{cases}$$

- $c = (c_1, \ldots, c_q)^T$ est le vecteur colonne (à q lignes) de calcul du coût ;
- $x = (x_1, \ldots, x_q)^T$ est le vecteurs colonnes (à q lignes) des variables ;
- $A = (a_{i,j})_{1 \leq i \leq p, 1 \leq j \leq q}$ est une matrice (à p lignes et q colonnes) des contraintes ;
- et $b = (b_1, \ldots, b_p)^T$ est le vecteur colonne (à p lignes) des bornes

Cela peut toujours se faire. L'astuce consiste à rajouter des variables (dites d'écart). Par exemple $x + 2y \leq 8$ sera remplacé par :

$\begin{cases} x + 2y + z & = 8 \\ z & \geq 0 \end{cases}$ $\rightsquigarrow$ forme (aussi) utilisée dans les librairies de résolution SciPy, cvxopt, PuLP ...

Simplexe : forme matricielle standard

Maximiser

$$c(x,y) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & \leq 8 \\ x & +y & \leq 5 \\ 9x & +4y & \leq 36 \\ & x,y & \geq 0 \end{cases}$$

Devient :

Maximiser

$$c(x,y,z_1,z_2,z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & = 8 \\ x & +y & +z_2 & = 5 \\ 9x & +4y & +z_3 & = 36 \\ & x,y, & z_1,z_2,z_3 & \geq 0 \end{cases}$$

$$A = \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 9 & 4 & 0 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 8 \\ 5 \\ 36 \end{pmatrix}$$

$$c = \begin{pmatrix} 2 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad X = \begin{pmatrix} x \\ y \\ z_1 \\ z_2 \\ z_3 \end{pmatrix}$$

Maximiser $\quad c^T X$

$$\text{sous les contraintes} \begin{cases} AX & = b \\ X & \geq 0 \end{cases}$$

Réseau et capacité

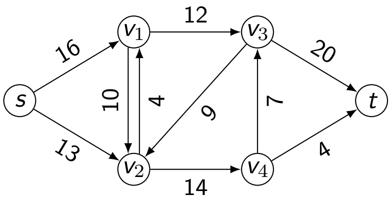
Un graphe de capacité (ou "réseau de transport") $G = (S, A)$ est un graphe **orienté** tel que :

- $\forall (u, v) \in A$, capacité $c(u, v) > 0$.
- Si $(u, v) \notin A$, on pose $c(u, v) = 0$
- présence de deux sommets particuliers :
  - s : "source" (pas d'arc entrants)
  - t : "puits" (pas d'arc sortants)

Réseau et capacité

Un graphe de capacité (ou "réseau de transport") $G = (S, A)$ est un graphe **orienté** tel que :

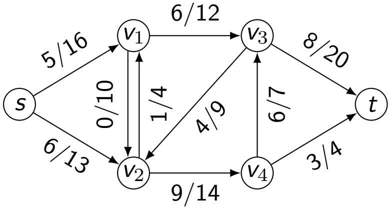
- $\forall (u, v) \in A$, capacité $c(u, v) > 0$.
- Si $(u, v) \notin A$, on pose $c(u, v) = 0$
- présence de deux sommets particuliers :
  - s : "source" (pas d'arc entrants)
  - t : "puits" (pas d'arc sortants)



Flot

Un *flot* est une fonction $f : S^2 \rightarrow \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u, v) \leq c(u, v)$
- **Anti-symétrie** $f(u, v) = -f(v, u)$
- **Conservation du flot** $\sum_{w \in S} f(u, w) = 0$, sauf si $u = s$ ou $u = t$



La *valeur* d'un flot est $\sum_{(s,u) \in A} f(s, u) = \sum_{(v,t) \in A} f(v, t)$

Flot maximum

Un problème classique est la recherche d'un flot maximum à partir d'un graphe de capacité.

Par exemple, si l'on considère que notre graphe de capacité représente un réseau de communication dont la capacité représente le débit. On voudrait connaitre le débit maximum pour envoyer des fichiers entre un émetteur (la source) et un récepteur (le puits).

Flot maximum

Un problème classique est la recherche d'un flot maximum à partir d'un graphe de capacité.

Par exemple, si l'on considère que notre graphe de capacité représente un réseau de communication dont la capacité représente le débit. On voudrait connaitre le débit maximum pour envoyer des fichiers entre un émetteur (la source) et un récepteur (le puits).

en R5.A12/R5.B.10 $\rightsquigarrow$ résolution par l'algorithme d'Edmond–Karp

Flot maximum

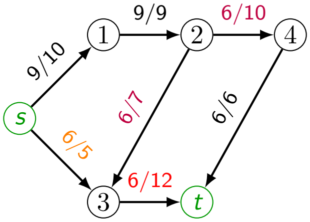
Un problème classique est la recherche d'un flot maximum à partir d'un graphe de capacité.

Par exemple, si l'on considère que notre graphe de capacité représente un réseau de communication dont la capacité représente le débit. On voudrait connaitre le débit maximum pour envoyer des fichiers entre un émetteur (la source) et un récepteur (le puits).

en R5.A12/R5.B.10 $\rightsquigarrow$ résolution par l'algorithme d'Edmond–Karp

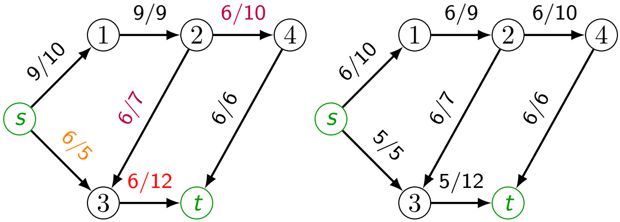
ici : résolution exacte par optimisation linéaire

Exemples de (non)-flots



Flot non valide

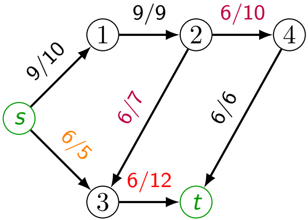
Exemples de (non)-flots



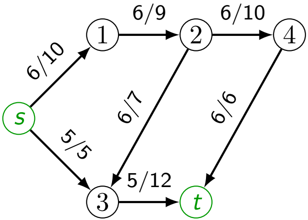
Flot non valide

Flot total $= 5 + 6 = 11$

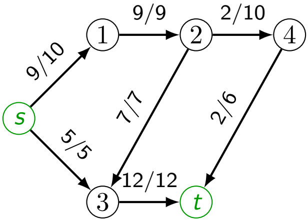
Exemples de (non)-flots



Flot non valide



Flot total $= 5 + 6 = 11$



Flot total

$= c(p_0) + c(p_1) + c(p_2) =$

14

$\rightsquigarrow$ flot maximum

Flot : résolution par le optimisation linéaire

Un *flot* est une fonction $f : S^2 \to \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u, v) \leq c(u, v)$
- **Anti-symétrie** $f(u, v) = -f(v, u)$
- **Conservation du flot** $\sum_{w \in S} f(u, w) = 0$, sauf si $u = s$ ou $u = t$

Flot : résolution par le optimisation linéaire

Un *flot* est une fonction $f : S^2 \to \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u, v) \leq c(u, v)$
- **Anti-symétrie** $f(u, v) = -f(v, u)$
- **Conservation du flot** $\sum_{w \in S} f(u, w) = 0$, sauf si $u = s$ ou $u = t$

$\rightsquigarrow$ problème d'optimisation linéaire :

Flot : résolution par le optimisation linéaire

Un *flot* est une fonction $f: S^2 \to \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u, v) \leq c(u, v)$
- **Anti-symétrie** $f(u, v) = -f(v, u)$
- **Conservation du flot** $\sum_{w \in S} f(u, w) = 0$, sauf si $u = s$ ou $u = t$

$\rightsquigarrow$ problème d'optimisation linéaire :

Maximiser $\sum_{(s,u) \in A} f(s, u)$

sous les contraintes

- $f(u, v) \leq c(u, v), \quad \forall (u, v) \in A$
- $f(u, v) \geq 0, \quad \forall (u, v) \in A$
- $\sum_{x \text{ voisin sortant de } u} f(u, x) - \sum_{y \text{ voisin entrant de } u} f(y, u) = 0$
- $\sum_{w \in S} f(u, w) = 0$

Flot : résolution par le optimisation linéaire

Un *flot* est une fonction $f : S^2 \to \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u, v) \leq c(u, v)$
- **Anti-symétrie** $f(u, v) = -f(v, u)$
- **Conservation du flot** $\sum_{w \in S} f(u, w) = 0$, sauf si $u = s$ ou $u = t$

$\rightsquigarrow$ problème d'optimisation linéaire :

Maximiser $\sum_{(s,u) \in A} f(s, u)$

sous les contraintes

- $f(u, v) \leq c(u, v), \quad \forall (u, v) \in A$
- $f(u, v) \geq 0, \quad \forall (u, v) \in A$
- $\sum_{x \text{ voisin sortant de } u} f(u, x) - \sum_{y \text{ voisin entrant de } u} f(y, u) = 0$
- $\sum_{w \in S} f(u, w) = 0$

variables $\rightsquigarrow$ valeurs du flot par arête

Flot : résolution par le optimisation linéaire

Maximiser $\sum_{(s,u)\in A} f(s,u)$

sous les contraintes

- $f(u,v) \leq c(u,v), \quad \forall (u,v) \in A$
- $f(u,v) \geq 0, \quad \forall (u,v) \in A$
- $\sum_{x \text{ voisin sortant de } u} f(u,x) - \sum_{y \text{ voisin entrant de } u} f(y,u) = 0$
- $\sum_{w\in S} f(u,w) = 0$

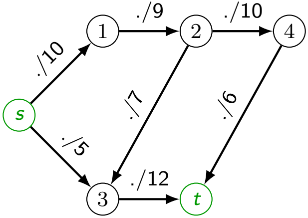
variables $\rightsquigarrow$ valeurs du flot par arête

$\rightsquigarrow$ construction

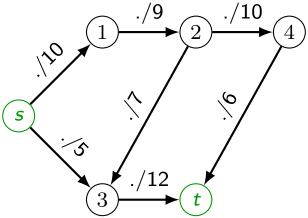
- une matrice A_{cons} qui garantira la conservation du flot
- un vecteur b_{cons}, égal à 0 sur toutes ses coordonnées
- une matrice A_{capa}, égale à la matrice identité
- un vecteur b_{capa} qui contiendra les capacités de chaque arête
- un vecteur de coût c qui vaut 1 sur les arêtes sortant de s

A_{cons} : matrice $nb_{\text{sommets}} - 2$ (la source et le puits) lignes et nb_{aretes} colonnes ($\equiv$ matrice d'incidence - lignes source et puits)

Flot max : optimisation linéaire



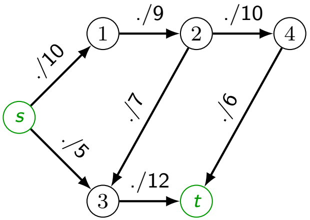
Flot max : optimisation linéaire



variables représentant les flux sur les arcs :

$$\begin{cases} x_1 = f_{s\to 1} \\ x_2 = f_{s\to 3} \\ x_3 = f_{1\to 2} \\ x_4 = f_{2\to 3} \\ x_5 = f_{2\to 4} \\ x_6 = f_{3\to t} \\ x_7 = f_{4\to t} \end{cases}$$

Flot max : optimisation linéaire



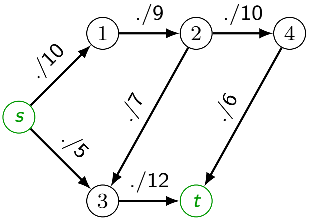
variables représentant les flux sur les arcs :

$$\begin{cases} x_1 = f_{s\to 1} \\ x_2 = f_{s\to 3} \\ x_3 = f_{1\to 2} \\ x_4 = f_{2\to 3} \\ x_5 = f_{2\to 4} \\ x_6 = f_{3\to t} \\ x_7 = f_{4\to t} \end{cases}$$

Contraintes de capacité

$$\begin{cases} x_1 \leq 10 \\ x_2 \leq 5 \\ x_3 \leq 9 \\ x_4 \leq 7 \\ x_5 \leq 10 \\ x_6 \leq 12 \\ x_7 \leq 6 \end{cases}$$

Flot max : optimisation linéaire



variables représentant les flux sur les arcs :

$$\begin{cases} x_1 = f_{s\to 1} \\ x_2 = f_{s\to 3} \\ x_3 = f_{1\to 2} \\ x_4 = f_{2\to 3} \\ x_5 = f_{2\to 4} \\ x_6 = f_{3\to t} \\ x_7 = f_{4\to t} \end{cases}$$

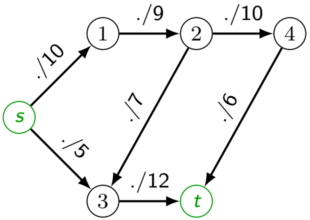
Contraintes de capacité

$$\begin{cases} x_1 \leq 10 \\ x_2 \leq 5 \\ x_3 \leq 9 \\ x_4 \leq 7 \\ x_5 \leq 10 \\ x_6 \leq 12 \\ x_7 \leq 6 \end{cases}$$

conservation du flux

$$\begin{cases} x_1 - x_3 = 0 & ① \\ x_3 - x_4 - x_5 = 0 & ② \\ x_2 + x_4 - x_6 = 0 & ③ \\ x_5 - x_7 = 0 & ④ \end{cases}$$

Flot max : optimisation linéaire



variables représentant les flux sur les arcs :

$$\begin{cases} x_1 = f_{s\to 1} \\ x_2 = f_{s\to 3} \\ x_3 = f_{1\to 2} \\ x_4 = f_{2\to 3} \\ x_5 = f_{2\to 4} \\ x_6 = f_{3\to t} \\ x_7 = f_{4\to t} \end{cases}$$

Contraintes de capacité

$$\begin{cases} x_1 \leq 10 \\ x_2 \leq 5 \\ x_3 \leq 9 \\ x_4 \leq 7 \\ x_5 \leq 10 \\ x_6 \leq 12 \\ x_7 \leq 6 \end{cases}$$

conservation du flux

$$\begin{cases} x_1 - x_3 = 0 & ① \\ x_3 - x_4 - x_5 = 0 & ② \\ x_2 + x_4 - x_6 = 0 & ③ \\ x_5 - x_7 = 0 & ④ \end{cases}$$

pblm :

$\max c = x_1 + x_2$

Flot max : optimisation linéaire

Vecteur des variables :

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix}$$

Vecteur fonction objectif :

$$\mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Vecteur des capacités maximales :

$$\mathbf{u} = \begin{bmatrix} 10 \\ 5 \\ 9 \\ 7 \\ 10 \\ 12 \\ 6 \end{bmatrix}$$

Matrice des contraintes de conservation $A\mathbf{x} = 0$:

$$A_{cons} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}$$

Matrice des capacité $A\mathbf{x} = 0$:

$$A_{capa} = I_7 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Résumé du problème :

$$\max_{\mathbf{x}} \quad \mathbf{c}^t\mathbf{x} = x_1 + x_2$$

sous les contraintes :

$$\begin{cases} A_{cons}\mathbf{x} = \mathbf{0} \\ \mathbf{0} \leq A_{capa}\mathbf{x} \leq \mathbf{u} \end{cases}$$

Bonus :

la suite ne sera pas exigible en contrôle, mais est intéressante pour aller plus loin dans le cours

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & = 8 \\ x & +y & +z_2 & & = 5 \\ 9x & +4y & +z_3 & & = 36 \\ & x, y, & z_1, z_2, z_3 & & \geq 0 \end{cases}$$

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & = 8 \\ x & +y & +z_2 & & = 5 \\ 9x & +4y & +z_3 & & = 36 \\ & x, y, & z_1, z_2, z_3 & & \geq 0 \end{cases}$$

On résout à partir de la solution

de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & = 8 \\ x & +y & +z_2 & & = 5 \\ 9x & +4y & +z_3 & & = 36 \\ & x, y, & z_1, z_2, z_3 & & \geq 0 \end{cases}$$

On résout à partir de la solution

de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & = 8 \\ x & +y & +z_2 & & = 5 \\ 9x & +4y & +z_3 & & = 36 \\ & x, y, & z_1, z_2, z_3 & & \geq 0 \end{cases}$$

On résout à partir de la solution de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Augmenter x ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 et z_3)

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & = 8 \\ x & +y & +z_2 & & = 5 \\ 9x & +4y & +z_3 & & = 36 \\ & x, y, & z_1, z_2, z_3 & & \geq 0 \end{cases}$$

On résout à partir de la solution de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Augmenter x ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 et z_3)

opt. $x = \frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y$

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & & = 8 \\ x & +y & & +z_2 & & = 5 \\ 9x & +4y & & & +z_3 & = 36 \\ & x, y, & z_1, z_2, z_3 & & & \geq 0 \end{cases}$$

On résout à partir de la solution de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Augmenter x ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 et z_3)

opt. $x = \frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y$

$$\begin{cases} z_1 & = 8 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -2y \\ z_2 & = 5 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -y \\ x & = \frac{36}{9} & -\frac{z_3}{9} & -\frac{4}{9}y \\ c & = & 2(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & +y \end{cases}$$

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & & = 8 \\ x & +y & & +z_2 & & = 5 \\ 9x & +4y & & & +z_3 & = 36 \\ & x, y, & z_1, z_2, z_3 & & & \geq 0 \end{cases}$$

On résout à partir de la solution de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Augmenter x ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 et z_3)

opt. $x = \frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y$

$$\begin{cases} z_1 & = 8 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -2y \\ z_2 & = 5 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -y \\ x & = \frac{36}{9} & -\frac{z_3}{9} & -\frac{4}{9}y \\ c & = & 2(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & +y \end{cases}$$

$$\begin{cases} z_1 & = 4 & +\frac{z_3}{9} & -\frac{14}{9}y \\ z_2 & = 1 & +\frac{z_3}{9} & -\frac{5}{9}y \\ x & = 4 & -\frac{z_3}{9} & -\frac{4}{9}y \\ c & = 8 & -\frac{2}{9}z_3 & +\frac{1}{9}y \end{cases}$$

Simplexe : pivotage et résolution I

Maximiser

$$c(x, y, z_1, z_2, z_3) = 2x + y$$

sous les contraintes

$$\begin{cases} x & +2y & +z_1 & & & = 8 \\ x & +y & & +z_2 & & = 5 \\ 9x & +4y & & & +z_3 & = 36 \\ & x, y, & z_1, z_2, z_3 & & & \geq 0 \end{cases}$$

On résout à partir de la solution de base

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 5, 36)$

(donc $c = 0$)

$$\begin{cases} z_1 & = 8 & -x & -2y \\ z_2 & = 5 & -x & -y \\ z_3 & = 36 & -9x & -4y \\ c & = & 2x & +y \end{cases}$$

Augmenter x ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 et z_3)

opt. $x = \frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y$

$$\begin{cases} z_1 & = 8 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -2y \\ z_2 & = 5 & -(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & -y \\ x & = \frac{36}{9} & -\frac{z_3}{9} & -\frac{4}{9}y \\ c & = & 2(\frac{36}{9} - \frac{z_3}{9} - \frac{4}{9}y) & +y \end{cases}$$

$$\begin{cases} z_1 & = 4 & +\frac{z_3}{9} & -\frac{14}{9}y \\ z_2 & = 1 & +\frac{z_3}{9} & -\frac{5}{9}y \\ x & = 4 & -\frac{z_3}{9} & -\frac{4}{9}y \\ c & = 8 & -\frac{2}{9}z_3 & +\frac{1}{9}y \end{cases}$$

avec comme solution

$(x, y, z_1, z_2, z_3) = (4, 0, 4, 1, 0)$

Simplexe : pivotage et résolution II

$$\begin{cases} z_1 & = 4 & + \frac{z_3}{9} & - \frac{14}{9} y \\ z_2 & = 1 & + \frac{z_3}{9} & - \frac{5}{9} y \\ x & = 4 & - \frac{z_3}{9} & - \frac{4}{9} y \\ c & = 8 & - \frac{2}{9} z_3 & + \frac{1}{9} y \end{cases}$$

Simplexe : pivotage et résolution II

$$\begin{cases} z_1 & = 4 & + \frac{z_3}{9} & - \frac{14}{9} y \\ z_2 & = 1 & + \frac{z_3}{9} & - \frac{5}{9} y \\ x & = 4 & - \frac{z_3}{9} & - \frac{4}{9} y \\ c & = 8 & - \frac{2}{9} z_3 & + \frac{1}{9} y \end{cases}$$

Augmenter y ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 ou x)

opt. $y = \frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2$

Simplexe : pivotage et résolution II

$$\begin{cases} z_1 & = 4 & + \frac{z_3}{9} & - \frac{14}{9} y \\ z_2 & = 1 & + \frac{z_3}{9} & - \frac{5}{9} y \\ x & = 4 & - \frac{z_3}{9} & - \frac{4}{9} y \\ c & = 8 & - \frac{2}{9} z_3 & + \frac{1}{9} y \end{cases}$$

Augmenter y ferait augmenter c,
donc on va l'augmenter un
maximum (ce qui fera diminuer
z_1, z_2 ou x)
opt. $y = \frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2$

$$\begin{cases} z_1 & = 4 & + \frac{z_3}{9} & - \frac{14}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \\ y & = \frac{9}{5} & + \frac{81}{5} z_3 & - \frac{9}{5} z_2 \\ x & = 4 & - \frac{z_3}{9} & - \frac{4}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \\ c & = 8 & - \frac{2}{9} z_3 & + \frac{1}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \end{cases}$$

Simplexe : pivotage et résolution II

$$\begin{cases} z_1 &= 4 &+\frac{z_3}{9} &-\frac{14}{9}y \\ z_2 &= 1 &+\frac{z_3}{9} &-\frac{5}{9}y \\ x &= 4 &-\frac{z_3}{9} &-\frac{4}{9}y \\ c &= 8 &-\frac{2}{9}z_3 &+\frac{1}{9}y \end{cases}$$

Augmenter y ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 ou x)

opt. $y = \frac{9}{5} + \frac{81}{5}z_3 - \frac{9}{5}z_2$

$$\begin{cases} z_1 &= 4 &+\frac{z_3}{9} &-\frac{14}{9}\left(\frac{9}{5} + \frac{81}{5}z_3 - \frac{9}{5}z_2\right) \\ y &= \frac{9}{5} &+\frac{81}{5}z_3 &-\frac{9}{5}z_2 \\ x &= 4 &-\frac{z_3}{9} &-\frac{4}{9}\left(\frac{9}{5} + \frac{81}{5}z_3 - \frac{9}{5}z_2\right) \\ c &= 8 &-\frac{2}{9}z_3 &+\frac{1}{9}\left(\frac{9}{5} + \frac{81}{5}z_3 - \frac{9}{5}z_2\right) \end{cases}$$

$$\begin{cases} z_1 &= \frac{6}{5} &+\frac{z_3}{5} &-\frac{14}{5}z_2 \\ y &= \frac{9}{5} &+\frac{81}{5}z_3 &-\frac{9}{5}z_2 \\ x &= \frac{16}{5} &-\frac{z_3}{5} &-\frac{4}{5}z_2 \\ c &= \frac{16}{5} &-\frac{1}{5}z_3 &-\frac{1}{5}z_2 \end{cases}$$

avec comme solution

$(x, y, z_1, z_2, z_3) = (\frac{16}{5}, \frac{9}{5}, \frac{6}{5}, 0, 0)$

Simplexe : pivotage et résolution II

$$\begin{cases} z_1 &= 4 &+ \frac{z_3}{9} &- \frac{14}{9} y \\ z_2 &= 1 &+ \frac{z_3}{9} &- \frac{5}{9} y \\ x &= 4 &- \frac{z_3}{9} &- \frac{4}{9} y \\ c &= 8 &- \frac{2}{9} z_3 &+ \frac{1}{9} y \end{cases}$$

Augmenter y ferait augmenter c, donc on va l'augmenter un maximum (ce qui fera diminuer z_1, z_2 ou x)

opt. $y = \frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2$

$$\begin{cases} z_1 &= 4 &+ \frac{z_3}{9} &- \frac{14}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \\ y &= \frac{9}{5} &+ \frac{81}{5} z_3 &- \frac{9}{5} z_2 \\ x &= 4 &- \frac{z_3}{9} &- \frac{4}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \\ c &= 8 &- \frac{2}{9} z_3 &+ \frac{1}{9} \left(\frac{9}{5} + \frac{81}{5} z_3 - \frac{9}{5} z_2 \right) \end{cases}$$

$$\begin{cases} z_1 &= \frac{6}{5} &+ \frac{z_3}{5} &- \frac{14}{5} z_2 \\ y &= \frac{9}{5} &+ \frac{81}{5} z_3 &- \frac{9}{5} z_2 \\ x &= \frac{16}{5} &- \frac{z_3}{5} &- \frac{4}{5} z_2 \\ c &= \frac{16}{5} &- \frac{1}{5} z_3 &- \frac{1}{5} z_2 \end{cases}$$

avec comme solution $(x, y, z_1, z_2, z_3) = (\frac{16}{5}, \frac{9}{5}, \frac{6}{5}, 0, 0)$

tous les coef. du coût négatifs $\rightsquigarrow$ plus d'augmentation possible $\rightsquigarrow$ stop

Simplexe : tableau et pivotage

	x	y	z_1	z_2	z_3		
z_1	1	2	1	0	0	8	$L'_1 \leftarrow L_1 - L'_3$
z_2	1	1	0	1	0	5	$L'_2 \leftarrow L_2 - L'_3$
z_3	$\boxed{9}$	4	0	0	1	36	$L'_3 \leftarrow L_3 \times \frac{1}{9}$
Max	2	1	0	0	0	0	$L'_4 \leftarrow L_4 - 2L'_3$

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 1, 36)$ (base)

Simplexe : tableau et pivotage

	x	y	z_1	z_2	z_3		
z_1	1	2	1	0	0	8	$L_1' \leftarrow L_1 - L_3'$
z_2	1	1	0	1	0	5	$L_2' \leftarrow L_2 - L_3'$
z_3	$\boxed{9}$	4	0	0	1	36	$L_3' \leftarrow L_3 \times \frac{1}{9}$
Max	2	1	0	0	0	0	$L_4' \leftarrow L_4 - 2L_3'$

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 1, 36)$ (base)

	x	y	z_1	z_2	z_3		
z_1	0	$\frac{14}{9}$	1	0	$-\frac{1}{9}$	4	$L_1' \leftarrow L_1 - \frac{14}{9} L_2'$
z_2	0	$\boxed{\frac{5}{9}}$	0	1	$-\frac{1}{9}$	1	$L_2' \leftarrow L_2 \times \frac{9}{5}$
x	1	$\frac{4}{9}$	0	0	$\frac{1}{9}$	4	$L_3' \leftarrow L_3 - \frac{4}{9} L_2'$
Max	0	$\frac{1}{9}$	0	0	$-\frac{2}{9}$	-8	$L_4' \leftarrow L_4 - \frac{1}{9} L_2'$

Simplexe : tableau et pivotage

	x	y	z_1	z_2	z_3		
z_1	1	2	1	0	0	8	$L_1' \leftarrow L_1 - L_3'$
z_2	1	1	0	1	0	5	$L_2' \leftarrow L_2 - L_3'$
z_3	$\boxed{9}$	4	0	0	1	36	$L_3' \leftarrow L_3 \times \frac{1}{9}$
Max	2	1	0	0	0	0	$L_4' \leftarrow L_4 - 2L_3'$

$(x, y, z_1, z_2, z_3) = (0, 0, 8, 1, 36)$ (base)

	x	y	z_1	z_2	z_3		
z_1	0	$\frac{14}{9}$	1	0	$-\frac{1}{9}$	4	$L_1' \leftarrow L_1 - \frac{14}{9} L_2'$
z_2	0	$\boxed{\frac{5}{9}}$	0	1	$-\frac{1}{9}$	1	$L_2' \leftarrow L_2 \times \frac{9}{5}$
x	1	$\frac{4}{9}$	0	0	$\frac{1}{9}$	4	$L_3' \leftarrow L_3 - \frac{4}{9} L_2'$
Max	0	$\frac{1}{9}$	0	0	$-\frac{2}{9}$	-8	$L_4' \leftarrow L_4 - \frac{1}{9} L_2'$

$(x, y, z_1, z_2, z_3) = (4, 0, 8, 5, 0)$

	x	y	z_1	z_2	z_3	
z_1	0	0	1	$-\frac{14}{5}$	$\frac{1}{5}$	$\frac{6}{5}$
y	0	1	0	$\frac{9}{5}$	$-\frac{1}{5}$	$\frac{9}{5}$
x	1	0	0	$-\frac{4}{5}$	$\frac{1}{5}$	$\frac{16}{5}$
Max	0	0	0	$-\frac{1}{5}$	$-\frac{1}{5}$	$-\frac{41}{5}$

$(x, y, z_1, z_2, z_3) = (3.2, 1.8, 1.2, 0, 0)$

On retrouve bien notre maximum

Simplexe : bilan

+ Raisonnable à implémenter
+ Efficace en pratique
– Besoin de règles supplémentaires pour être meilleur
– Exponentiel en pire cas
– Attention à la précision si on divise par des petits nombre (idem pivot)

Simplexe : bilan

+ Raisonnable à implémenter
+ Efficace en pratique
– Besoin de règles supplémentaires pour être meilleur
– Exponentiel en pire cas
– Attention à la précision si on divise par des petits nombre (idem pivot)

Règles de pivotage : critère de Dantzig
En rajoutant des (petites) perturbations aléatoires, on converge presque sûrement vers une solution proche de la solution réelle en temps polynomial
Il existe aussi des algo. réellement polynomiaux (mais on ne sait pas s'il existe une règle de pivotage qui rend le simplexe polynomial)