Prof. F. Eisenbrand EPFL - DISOPT

Optimisation Discrète

Adrian Bock Semestre de printemps 2011

Série 6 — 31 mars 2011

Remarque générale :
Pour obtenir un bonus pour l'évaluation finale, vous pouvez rendre des solutions écrites à l'exercice noté avant (le début de) la séance de la semaine prochaine. Les mathématiciens doivent traiter l'exercice (*). Les autres par contre peuvent choisir entre l'exercice (*) ou (Δ).
Le rendu peut être fait en groupe de trois personnes au plus.

Exercice 1
Trouver la forme duale des programmes linéaires suivants :

(i)

$$\begin{array}{rrcl}
\max & 2x_1 + 3x_2 - 7x_3 & & \\
 & x_1 + 3x_2 + 2x_3 & = & 4 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 - x_3 & \geq & -15 \\
 & x_1, x_2 & \geq & 0
\end{array}$$

(ii)

$$\begin{array}{rrcr}
\min & 3x_1 + 2x_2 - 3x_3 + 4x_4 & & \\
 & 2x_1 - 2x_2 + 3x_3 + 4x_4 & \leq & 3 \\
 & x_2 + 3x_3 + 4x_4 & \geq & -5 \\
 & 2x_1 - 3x_2 - 7x_3 - 4x_4 & = & 2 \\
 & x_1 \geq 0 & & \\
 & x_4 \leq 0 & &
\end{array}$$

Solution

(i)

$$\begin{array}{rrrrcr}
\min & 4y_1 & 8y_2 & -15y_3 & & \\
 & y_1 & +y_2 & +y_3 & \geq & 2 \\
 & 3y_1 & +y_2 & & \geq & 3 \\
 & 2y_1 & & -y_3 & = & -7 \\
 & & y_2 & & \geq & 0 \\
 & & & y_3 & \leq & 0
\end{array}$$

(ii)

$$\begin{array}{lrcr} \max & 3y_1 - 5y_2 + 2y_3 & & \\ & 2y_1 + 2y_3 & \leq & 3 \\ & -2y_1 + y_2 - 3y_3 & = & 2 \\ & 3y_1 + 3y_2 - 7y_3 & = & -3 \\ & y_2 + y_2 - y_3 & \geq & 1 \\ & y_1 \leq 0 & & \\ & y_2 \geq 0 & & \end{array}$$

Exercice 2
Soit PL un programme linéaire admissible et borné. Donner une preuve ou un contre-exemple

(i) pour chaque entrée du tableau suivant

$\Longrightarrow$	$\exists!$ toit optimal	$\exists!$ solution optimale
PL est dégénéré	*	*
PL est non-dégénéré	*	*

(ii) pour l'affirmation :
Il est impossible qu'un programme linéaire admette une infinité de solutions optimales sous la forme primale et duale.

Solution

(i) pour chaque entrée du tableau suivant

$\Longrightarrow$	$\exists!$ toit optimal	$\exists!$ solution optimale
PL dégénéré	non	non
PL non-dégénéré	non	oui

L'exercice 3 de la série 5 donne un contre-exemple pour un PL dégénéré où l'on a plusieurs toits optimaux et solutions optimales. On peut trouver plusieurs toits optimaux même pour un PL non-dégénéré, comme figuré ci-dessous. Mais on n'a qu'une unique solution optimale par la définition du cas non-dégénéré.

(ii) Non. Contre-exemple :

$$\begin{array}{lrcl} \max & 0 & & \\ & x_1 + x_2 & = & 0 \\ & x_1 + x_2 & = & 0 \end{array} \qquad \begin{array}{lrcl} \min & 0 & & \\ & y_1 + y_2 & = & 0 \\ & y_1 + y_2 & = & 0 \end{array}$$

Exercice 3 ($*$)

Soit A une matrice de taille $m \times n$ et $b \in \mathbb{R}^m$ un vecteur.

Démontrer que le polyèdre $P := \{x \in \mathbb{R}^n \colon Ax \leq b\}$ est vide si et seulement s'il existe $\lambda \geq 0$ tel que $\lambda^T A = 0$ et $\lambda^T b = -1$.

Indications :
– Considérer d'abord le cas où A est de plein rang-colonne et ensuite le cas général.
– Utiliser la dualité.

Solution

$\Rightarrow$ Supposons que le polyèdre P est vide. Considérons le programme linéaire qui maximise l'objectif 0 parmi les points dans P. Par construction, ce PL n'est pas admissible. Son PL dual est

$$\min\{b^T y \colon A^T y = 0, \quad y \geq 0\}$$

Comme 0 est une solution admissible, le PL dual est non-borné par la dualité (sinon le PL primal serait admissible). En effet, il existe $y' >= 0$ tel que $A^T y' = 0$ et $b^T y' < 0$. Donc $\lambda = \frac{y'}{|b^T y'|}$ est le certificat voulu, car $\lambda^T A = \frac{1}{|b^T y'|} A^T y' = 0$ et $b^T \lambda = \frac{b^T y'}{|b^T y'|} = -1$

$\Leftarrow$ Étant donné $\lambda \geq 0$ avec $\lambda^T = 0$ et $\lambda^T b = -1$, supposons que P n'est pas vide. Alors il existe $x \in P$ tel que $Ax \leq b$. Comme λ est un vecteur positif, on a donc $\lambda^T Ax \leq \lambda^T b$, c'est-à-dire $0 \leq -1$, une contradiction.

Exercice 4 (Δ)

Considérer le programme linéaire suivant

$$\begin{array}{lrcl}
\max & 10x_1 + 8x_2 + 6x_3 + 9x_4 + 18x_5 + 20x_6 + 15x_7 + 17x_8 + 15x_9 + 16x_{10} + 13x_{11} + 17x_{12} \\
 & 5x_1 + 6x_5 + 13x_9 & \leq & 2100 \\
 & 7x_2 + 12x_6 + 14x_{10} & \leq & 2100 \\
 & 4x_3 + 8x_7 + 9x_{11} & \leq & 2100 \\
 & 10x_4 + 15x_8 + 17x_{12} & \leq & 2100 \\
 & x_1 + x_2 + x_3 + x_4 & \geq & 100 \\
 & x_5 + x_6 + x_7 + x_8 & \geq & 150 \\
 & x_9 + x_{10} + x_{11} + x_{12} & \geq & 100 \\
 & x_1, \ldots, x_{12} & \geq & 0
\end{array}$$

Donner sa forme duale et trouver une solution optimale (SAGE !). Certifier l'optimalité de la solution obtenue en donnant une solution duale qui est admissible et atteint la valeur optimale.

Solution
Le PL dual est

$$
\begin{array}{rrcl}
\min & \multicolumn{3}{l}{2100 * (y_1 + y_2 + y_3 + y_4) - 100 * y_5 - 150 * y_6 - 100 * y_7} \\
& 5y_1 - y_5 & \geq & 10 \\
& 7y_2 - y_5 & \geq & 8 \\
& 4y_3 - y_5 & \geq & 6 \\
& 10y_4 - y_5 & \geq & 9 \\
& 6y_1 - y_6 & \geq & 18 \\
& 12y_2 - y_6 & \geq & 20 \\
& 8y_3 - y_6 & \geq & 15 \\
& 15y_4 - y_6 & \geq & 17 \\
& 13y_1 - y_7 & \geq & 15 \\
& 14y_2 - y_7 & \geq & 16 \\
& 9y_3 - y_7 & \geq & 13 \\
& 17y_4 - y_7 & \geq & 17 \\
& y_1, \ldots, y_7 & \geq & 0
\end{array}
$$

Le code SAGE suivant donne les deux PL.

```
p = MixedIntegerLinearProgram(maximization=True)
w = p.new_variable(integer=False)
p.add_constraint(5*w[0]+6*w[4]+13*w[8]<= 2100)
p.add_constraint(7*w[1]+12*w[5]+14*w[9]<= 2100)
p.add_constraint(4*w[2]+8*w[6]+9*w[10]<= 2100)
p.add_constraint(10*w[3]+15*w[7]+17*w[11]<= 2100)
p.add_constraint(w[0] + w[1] + w[2] + w[3] >= 100)
p.add_constraint(w[4] + w[5] + w[6] + w[7] >= 150)
p.add_constraint(w[8] + w[9] + w[10] + w[11] >= 100)
p.set_objective(10*w[0] + 8*w[1] + 6*w[2] + 9 * w[3] +18*w[4] +\
20*w[5]+15*w[6]+17*w[7]+15*w[8]+16*w[9]+13*w[10]+17*w[11])
p.solve()

d = MixedIntegerLinearProgram(maximization=False)
v = d.new_variable(integer = False)
d.add_constraint(5*v[0] -v[4] >= 10)
d.add_constraint(7*v[1] -v[4] >= 8)
d.add_constraint(4*v[2] -v[4] >= 6)
d.add_constraint(10*v[3] -v[4] >= 9)
d.add_constraint(6*v[0] -v[5] >= 18)
d.add_constraint(12*v[1] -v[5] >= 20)
d.add_constraint(8*v[2] -v[5] >= 15)
d.add_constraint(15*v[3] -v[5] >= 17)
d.add_constraint(13*v[0] -v[6] >= 15)
d.add_constraint(14*v[1] -v[6] >= 16)
d.add_constraint(9*v[2] -v[6] >= 13)
d.add_constraint(17*v[3] -v[6] >= 17)
d.set_objective(2100 * v[0] + 2100 * v[1] + 2100*v[2] + \
2100*v[3]-100*v[4]-150*v[5]-100*v[6])
d.show()
d.solve()
```

On trouve des solutions optimales pour les deux PL :
$x^* = (0, 0, 100, 0, 350, 175, 425/2, 80/3, 0, 0, 0, 100)$ et
$y^* = (3, 5/3, 15/8, 17/15, 3/2, 0, 34/15)$