Séminaire Mathématique de Béjaïa (LaMOS)
Volume 6, 2008, pp. 3 - 8

1

Résolution d'un problème de programmation bi-niveaux linéaire par la méthode DC

Aicha ANZI et Mohammed Said RADJEF

Laboratoire de Modélisation et d'Optimisation des Systèmes (LAMOS)
Université de Béjaïa, Béjaïa 06000, Algérie
Tél. (213) 34 21 51 88

Résumé Dans ce travail, nous traitons un problème de programmation bi-niveaux linéaire. La méthode utilisée est l'optimisation DC ainsi que l'algorithme DCA ; des techniques d'optimisation non convexe. Une étude de simulation a été réalisée et les résultats numériques sont présentés.

Mots clés :problème bi-niveaux, méthode DC, DCA, linéaire.

1.1 Introduction

Le problème de programmation bi-niveaux est un problème d'optimisation hiérarchique, avec deux niveaux de décision. Le premier niveau est appelé Leader et le deuxième Suiveur. Un programme bi-niveaux linéaire (PBL) possède la forme suivante :

$$\begin{cases} \max\limits_{x} \; F(x,y) = c_1^t x + d_1^t y, \\ \quad s.c. \;\; A_1 x + B_1 y \leq b_1, \\ \qquad\quad x \geq 0; \\ \qquad\quad \max\limits_{y} \; f(x,y) = c_2^t x + d_2^t y, \\ \qquad\qquad s.c. \;\; A_2 x + B_2 y \leq b_2, \\ \qquad\qquad\quad y \geq 0, \end{cases} \tag{1.1}$$

où x, c_1, $c_2 \in \mathbb{R}^{n_1}$; y, d_1, $d_2 \in \mathbb{R}^{n_2}$; $b_1 \in \mathbb{R}^{m_1}$; $b_2 \in \mathbb{R}^{m_2}$; $A_1 \in \mathbb{R}^{m_1 \times n_1}$; $B_1 \in \mathbb{R}^{m_1 \times n_2}$; $A_2 \in \mathbb{R}^{m_2 \times n_1}$ et $B_2 \in \mathbb{R}^{m_2 \times n_2}$.

Les définitions suivantes sont liées au problème (1.1) [4] :

a. *Ensemble des contraintes du problème*

$$S = \{(x,y) : A_1 x + B_1 y \leq b_1, A_2 x + B_2 y \leq b_2, x \geq 0, y \geq 0\}.$$

b. *Ensemble réalisable du Suiveur pour un x fixé*

$$S(x) = \{y \in \mathbb{R}^{n_2} : B_2 y \leq b_2 - A_2 x, y \geq 0\}.$$

c. *Projection de S sur l'espace du Leader*

$$P(X) = \left\{ x \in \mathbb{R}^{n_1} : \exists y \in \mathbb{R}^{n_2}, A_1 x + B_1 y \leq b_1, A_2 x + B_2 y \leq b_2, x \geq 0, y \geq 0 \right\}.$$

d. *Ensemble des réactions rationnelles du Suiveur pour $x \in P(X)$*

$$R(x) = \left\{ y \in \mathbb{R}^{n_2} : y = \arg\max[f(x, \hat{y}) : \hat{y} \in S(x)] \right\}.$$

e. *Région induite*

$$RI = \left\{ (x, y) \in S,\ y \in R(x) \right\}.$$

La région induite représente l'ensemble sur lequel le Leader optimise sa fonction objectif.

Bien que les fonctions objectif et les contraintes des deux niveaux sont linéaires, les problèmes de programmation bi-niveaux linéaires sont NP-difficiles et non convexes.

Dans ce travail nous traitons ces problèmes par une technique d'optimisation non convexe basée sur la programmation DC et l'algorithme DCA. Cette technique traite les problèmes d'optimisation où la fonction objectif peut être représentée sous la forme de différence de deux fonctions convexes

$$f(x) = g(x) - h(x)$$

où g et h sont appelées les composantes DC de f et $g - h$ la décomposition DC de f. Un programme DC s'écrit donc

$$\alpha = \inf\{ f(x) = g(x) - h(x) \ : \ x \in X \}, \tag{1.2}$$

1.2 DCA pour la résolution de $(\boldsymbol{PBL})$

1.2.1 Reformulation via une pénalité exacte

Soit le problème (1.1). En remplaçant le problème du deuxième niveau par ses conditions d'optimalité de Karush-Kuhn-Tucker on obtient :

$$\begin{cases} \max\limits_{x,y} \ F(x, y) = c_1^t x + d_1^t y \\ \qquad\quad A_1 x + B_1 y \leq b_1 \\ \qquad\quad A_2 x + B_2 y + w = b_2 \\ \qquad\quad B_2^t u - v = d_2 \\ \qquad\quad v^t y + u^t w = 0 \\ \qquad\quad x \geq 0,\ y \geq 0,\ u \geq 0,\ v \geq 0,\ w \geq 0, \end{cases} \tag{1.3}$$

En passant par les transformations suivantes
$z = \begin{pmatrix} x \ y \ e \ w \ v \ u \end{pmatrix}^t \in \mathbb{R}^n, \quad c = \begin{pmatrix} -c_1 \ -d_1 \ 0 \ 0 \ 0 \ 0 \end{pmatrix}^t \in \mathbb{R}^n,$

$$A = \begin{pmatrix} A_1 & B_1 & I_{m_1} & 0 & 0 & 0 \\ A_2 & B_2 & 0 & I_{m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & -I_{n_2} & B_2^t \end{pmatrix} \in R^{m\times n}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ d_2 \end{pmatrix} \in \mathbb{R}^m,$$

$E_u = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & I_{m_2} \end{pmatrix}, \quad E_v = \begin{pmatrix} 0 & 0 & 0 & 0 & I_{n_2} & 0 \end{pmatrix},$
$E_w = \begin{pmatrix} 0 & 0 & 0 & I_{m_2} & 0 & 0 \end{pmatrix}, \quad E_y = \begin{pmatrix} 0 & I_{n_2} & 0 & 0 & 0 & 0 \end{pmatrix},$
où
I_k est une $k \times k$ matrice identité ; 0 est la matrice nulle avec la dimension appropriée pour chaque cas, avec $n = n_1 + 2n_2 + m_1 + 2m_2$; $m = m_1 + m_2 + n_2$.
En utilisant ces notations, nous aurons :
$u^t w = (E_u z)^t (E_w z) = z^t (E_u^t E_w) z = z^t D^1 z$, et $v^t y = (E_v z)^t (E_y z) = z^t (E_v^t E_y) z = z^t D^2 z$,
ce qui donne : $u^t w + v^t y = z^t D^1 z + z^t D^2 z = z^t D z$ avec $D^1 + D^2 = D$.
Notons que les éléments $d_{ij} (i = \overline{1,n},\ j = \overline{1,n})$ de la matrice D sont tous non négatifs. En posant $Dz = q(z)$, on obtient le problème :

$$\min \left\{ F(z) = c^t z,\ Az = b,\ z^t q(z) = 0,\ z \geq 0 \right\}. \tag{1.4}$$

avec $q(z) \geq 0,\ \forall z \geq 0$.
Considérons l'ensemble convexe $\mathcal{Z} = \{z \in \mathbb{R}^n : Az = b, z \geq 0\}$, et soit la fonction $\Psi : \mathbb{R}^n \to \mathbb{R}$ définie par $\Psi(z) = \sum_{i=1}^{n} \min\{q_i(z),\ z_i\}$.
Ψ est une fonction concave, finie et non négative sur $\mathcal{Z}$. Nous avons

$$\{z \in \mathcal{Z},\ z^t q(z) = 0\} = \{z \in \mathcal{Z},\ \Psi(z) \leq 0\}.$$

Le problème $P(z)$ peut être réécrit sous la forme

$$\alpha = \min\{F(z)\ :\ z \in \mathcal{Z},\ \Psi(z) \leq 0\}. \tag{1.5}$$

D'après ([1], *théorème 1*), si $\mathcal{Z}$ est non vide et borné, alors il existe une constante $k_0 \geq 0$, telle que pour tout $k \geq k_0$ le problème (1.5) est équivalent au problème pénalisé

$$\alpha(k) = \min\{F(z) + k\Psi(z) : z \in \mathcal{Z}\}. \tag{1.6}$$

1.2.2 Décomposition DC de (1.6)

Soit $\chi_{\mathcal{Z}}$ la fonction indicatrice de l'ensemble $\mathcal{Z}$. Soient g et h deux fonctions définies par

$$g(z) = \chi_{\mathcal{Z}}(z) \ \text{ et } \ h(z) = -F(z) - k\Psi(z). \tag{1.7}$$

Donc g et h sont convexes et le problème (1.6) est un programme DC de la forme

$$\min\{g(z) - h(z) : z \in \mathbb{R}^n\}. \tag{1.8}$$

L'application de DCA pour le problème (1.8) revient à calculer les deux séquences $\{t^i\}$ et $\{z^i\}$ définies par

$$t^i \in \partial h(z^i) \ \text{ et } \ z^{i+1} \in \partial g^*(t^i).$$

En utilisant les règle de calcul en analyse convexe, on obtient :
• $t^i = -c + k\theta^i$, avec

$$\theta^i = -\sum_{j=1}^{n} \begin{cases} D_j^t, & si \;\; z_j^i > D_j z^i, \\ e_j, & si \;\; z_j^i < D_j z^i, \\ \gamma e_j + (1-\gamma) D_j^t, & si \;\; z_j^i = D_j z^i, \end{cases} \tag{1.9}$$

où D_j est la *j-ème* ligne de D, e_j est le *j-ème* vecteur unitaire de $\mathbb{R}^n$ et $\gamma \in [0,1]$.
• z^{i+1} est solution du problème

$$\min\{-\langle z,\ t^i\rangle : z \in \mathcal{Z}\}, \tag{1.10}$$

DCA pour (1.6)

1 : Soit z^0 point initial et $\epsilon > 0$. Poser $i = 0$, $k \in \mathbb{R}_+$, $\gamma \in [0,1]$ et $\lambda > 0$.
2 : Calculer $t^i \in \partial h(z^i)$ via (1.9).
3 : Calculer $z^{i+1} \in \partial g^*(t^i)$ en résolvant (1.10).
4 : Si $y^{i+1} \in \arg\max\left\{f(x^{i+1}, y) : B_2 y \leq b_2 - A_2 x^{i+1},\ y \geq 0\right\}$, **alors** aller à **5** ;
sinon aller à **7**.
5 : Calculer (v^*, u^*), solution du problème
$\min\{u^t(b_2 - A_2 x^{i+1}) : B_2^t u - v = d_2,\ u \geq 0,\ v \geq 0\}$, donc
$z^{i+1} = (x^{i+1}, y^{i+1}, e^{i+1}, w^{i+1}, v^*, u^*)$.
6 : Si $\|z^{i+1} - z^i\|/(\|z^i\| + 1) \leq \epsilon$, **alors** stop z^{i+1} est solution optimale de (1.1) ;
sinon aller à **7**.
7 : Poser $z^i = z^{i+1}$, $i = i+1$, $k = k + \lambda$ et aller à **2**.

1.2.3 Remarques

i) Le problème (1.6), avec la décomposition (1.7), est un programme DC polyédral, puisque $g(z) = \chi_{\mathcal{Z}}(z)$ est polyédrale [7]. Donc DCA appliqué à (1.6) possède une convergence finie [2][3].
ii) Dans l'étape 4 on teste la faisabilité de la solution (x, y) pour le (PBL). La vérification de ce test implique que $y \in R(x)$. Et puisque $(x, y) \in S$, nous aurons alors $(x, y) \in RI$ ce qui implique que (x, y) est une solution réalisable du (PBL).
iii) L'algorithme DCA calcule la solution optimale $z^* = (x^*, y^*, e^*, w^*, v^*, u^*)$ du problème (1.6). Si on définit l'ensemble Z_r comme

$$Z_r = \{z = (x, y, w, e, v, u) \in \mathbb{R}^n : z \in \mathcal{Z}, (x, y) \in RI\},$$

alors, d'après *ii*), $z^* \in Z_r$. Une telle solution n'est pas forcément une solution du problème (1.1) à cause de la présence des composantes duales (v^*, u^*). Le théorème suivant est inspiré de [5] et nous donne une condition pour qu'une solution z^* soit solution de (1.1).

Théorème 1.1 *Soit $k \in \mathbb{R}_+$ et soit l'ensemble $\bar{Z} \subset \mathcal{Z}$ défini par*

$$\bar{Z} = \{z \in \mathbb{R}^n : z \in Z_r,\ \exists I \subset \{1, \ldots, n\}/z_i = 0, \forall i \in I; q_i(z) = 0, \forall i \notin I\},$$

Une solution optimale z^* du problème (1.6) est solution de (1.1), si $z^* \in \bar{Z}$.

1.3 Résultats numériques

Pour évaluer les performances de notre algorithme nous l'avons implémenté sous MATLAB et testé sur un ensemble de 15 problèmes dont la solution optimale est connue. Le point initial a été calculé en résolvant le problème

$$0 = \min\{\Psi(z) : \bar{A}z = \bar{b}, z \geq 0\} \tag{1.11}$$

où

$$\bar{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & B_2 & 0 & I_{m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & -I_{n_2} & B_2^t \end{pmatrix} \quad \text{et} \quad \bar{b} = \begin{pmatrix} 0 \\ b_2 \\ d_2 \end{pmatrix}.$$

Nous avons pris $\gamma = 0.5$ et $\epsilon \leq 10^{-3}$.

Dans le but de faire une comparaison, nous avons implémenté l'algorithme développé par Y. Lv et al. [6] pour les problèmes de programmation bi-niveaux linéaire sans contraintes du Leader.

Les résultats numériques ont montré que notre algorithme est rapide, le temps d'exécution est petit et la procédure du calcul du point de départ est efficace car, avec un bon choix d u paramètre de pénalité, il converge souvent vers la solution globale.

Concernant la comparaison avec l'autre algorithme, nous avons remarqué que les résultats sont comparables. Cependant, nous avons constaté quelques points de différence concernant essentiellement la sensibilité des deux algorithmes aux paramètres d'entrée.

1.4 Conclusion

Nous avons présenté un algorithme d'optimisation DC pour la résolution du problème de programmation bi-niveaux linéaire, où le problème du Suiveur est remplacé par les conditions d'optimalité KKT associées. Le problème est ensuite reformulé comme un programme DC à l'aide d'une pénalité exacte et une décomposition DC adéquate. L'algorithme proposé est simple et rapide, puisque seuls des programmes linéaires sont résolus à chaque itération. Les résultats numériques montrent, qu'avec un bon choix du paramètre de pénalité, qui dépend du problème testé, l'algorithme converge souvent vers la solution globale.

Références

1. L.T.H. An and P.D. Tao : A continuous approach for globally solving linearly constrained quadratic zero-one programming problems. Optimization, 50 : 93–120, (2001)
2. L.T.H. An and P.D. Tao : Convex analysis approach to dc programming : theory and applications. Acta Mathematica Vietnamica, 22 : 289-355, (1997)
3. L.T.H. An and P.D. Tao : The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems. Annals Oper. Res., 133 :23–46, (2005)
4. J.F. Bard : Practical bilevel optimization : algorithms and applications. Kluwer academic publishers, Dordrecht (1998)

5. S. Dempe and A.G. Mersha. Linear bilevel programming with upper level constraints depending on the lower level solution. App. Math. and Comp., 180 : 247-254 (2006)
6. Y. Lv, T. Hu, G. Wang and Z. Wan. A penalty function method based on Kuhn–Tucker condition for solving linear bilevel programming. App. Math. and Comp., (2006)
7. R.T. Rockafellar : Convex analysis. Princeton, USA (1970)