Vietnam Journal of Mathematics **29**:3 (2001) 235–256

Vietnam Journal of MATHEMATICS

On Penalty Function Method for a Class of Nonconvex Constrained Optimization Problems*

Nguyen Van Quy and Le Dung Muu

Institute of Mathematics, P. O. Box 631, Bo Ho, Hanoi, Vietnam

Received January 31, 2000
Revised June 9, 2000

Abstract. We study penalty function method for dual form of a class of nonconvex mathematical programming problems which contains optimization problems over the efficient and weakly efficient sets, and linear bilevel programs as special cases. In contrast to the primal forms the resulting penalized problems for the dual form allows handling the dual variables of the problem whose effective domains of the objective function as well as the constraints are given explicitly. Application to linear bilevel programming is considered.

1. Introduction

Recently some nonconvex mathematical programming problems, whose feasible domain is the solution-set of another optimization problem, have been considered intensively. Examples for these problems are linear bilevel programs, optimization over the efficient set and weakly efficient set of a multiple objective programming problem. These problems have some important applications in decision making and different fields of world real life. Mathematically, they are difficult nonconvex constrained global optimization problems because their feasible domains in general are neither convex nor given explicitly.

There are several ways to formulate these problems among them the primal and dual formulations are widely used. The primal formulations deal only with

*This paper was supported in part by the National Basic Research Program in Natural Sciences, Vietnam.

the primal variables whereas the dual formulations use both the primal and dual variables.

To avoid difficulties arisen from the fact that the feasible domain is nonconvex and not given explicitly in a traditional format for an ordinary mathematical programming problem, several penalty function methods have been developed for the primal forms. In the articles [1, 2, 5, 8, 12, 22] penalty function methods have been developed for primal forms of optimization over the efficient set and weakly efficient set. In [8] exact penalty function methods have been considered for concave minimization subject to linear constraints and an additional facial reverse convex constraint. It has been shown in [8] that, among others, the primal forms of concave minimization over the efficient set of a multiple objective linear program and linear bilevel programming problems are of this type.

In this paper we study penalty function methods concerning dual forms for a class of nonconvex optimization problems which contains optimization problems over the efficient and weakly efficient sets, and linear bilevel program as special cases. We make use of the exact penalty function to develop an algorithm for solving a linear bilevel programming problem which computes an exact penalty parameter iteratively. In an important special case we give an estimation for the exact penalty parameter. For linear optimization over the efficient set and linear bilevel problem our study can be considered as a parallel work of that of Fülöp in [8]. The main difference between two approaches is that here we use the dual (parametric) forms of the problems rather than the primal forms. Comparing with the primal forms the dual forms have the advantages that they allows us to handle dual variables which in general is much less than the primal variables. For global optimization problems this is essential, since it is well known that computational costs (memory, time...) for solving a global optimization problem increase very quickly as the dimension of the problem gets larger. Moreover in contrast to the primal forms, in the dual forms the effective domains of the objective function as well as the constraints are given explicitly. When applying basic techniques such as branch-and-bound and outer approximation for solving a global optimization problem, in general, it requires constructing at the beginning a simple structured set (box, simplex, polyhedral cone) containing a solution and contained in the domain where the objective functions and constraints are finite. In the case where this set is not given explicitly, constructing such a simple set in general is not an easy task.

2. The Problem Statement and Examples

Consider the following problem

$$\alpha_* := \min\{f(x) \mid h(\lambda, x) = 0, \lambda \in \Lambda, x \in X\}, \tag{P}$$

where $\Lambda \subset R^p$, $X \subset R^n$ are polyhedral convex sets, $f : X \to R$, and $h : \Lambda \times X \to R$. Throughout this paper we assume that Λ, X are bounded and that $h(\lambda, x) \geq 0$ for every $(\lambda, x) \in \Lambda \times X$.

Let D denote the projection on R^n of the feasible domain of Problem (P), i.e.,

$$D := \{x \in X \mid \exists\lambda \in \Lambda, h(\lambda, x) = 0\}.$$

Clearly (P) is equivalent to the problem

$$\alpha_* := \min\{f(x) \mid x \in D\} \tag{P'}$$

in the sense that if (λ, x) is a global optimal solution to (P), then x is a global optimal solution to (P'), and if x is a global optimal solution of (P') then for every $\lambda \in \Lambda$ satisfying $h(\lambda, x) = 0$, the point (λ, x) is a global optimal solution to (P).

Below are examples for Problems (P). In what follows we write ab or $\langle a, b\rangle$ for the inner product of two vectors a and b.

2.1. Optimization Over the Efficient Set and Weakly Efficient Set

Let $F : X \to R^p$ be an affine fractional vector valued function. We recall that a point $x \in X$ is said to be efficient (resp. weakly efficient) of the vector optimization problem

$$\text{vmin}\{F(x) \mid x \in X\} \tag{VP}$$

if there does not exist $y \in X$ such that $F(y) \leq F(x)$, $F(y) \neq F(x)$ (resp. $F(y) < F(x)$). As usual here and below for two vectors $a = (a_1, \dots, a_p)$ and $b = (b_1, \dots, b_p)$, the notation $a \leq b$ (resp. $a < b$) means that $a_i \leq b_i$ (resp. $a_i < b_i$) for every i. We will denote by $E(F, X)$ and $WE(F, X)$ the sets of efficient points and weakly efficient points of (VP), respectively. The minimization problems over the efficient set and weakly efficient set of (VP) to be considered in this paper can be given respectively as

$$\min\{f(x) \mid x \in E(F, X)\} \tag{0}$$

and

$$\min\{f(x) \mid x \in WE(F, X)\}, \tag{1}$$

where f is a given real valued function defined on X.

Suppose that the affine fractional function F has the following form

$$F(x) = \Big(\frac{A_1x + s_1}{B_1x + t_1}, \dots, \frac{A_px + s_p}{B_px + t_p}\Big),$$

where A_i, B_i are n-dimensional row vectors, s_i, t_i are real numbers for all ($i = 1, \dots, p$). As usual we assume that $B_ix + t_i > 0$ for all $x \in X$ and all $i = 1, \dots, p$.

Let

$$S_0 := \{\lambda = (\lambda_1, \dots, \lambda_p) \mid \lambda > 0, \sum_{j=1}^{p} \lambda_j = 1\},$$

and

$$S_1 := \{\lambda = (\lambda_1, \dots, \lambda_p) \mid \lambda \geq 0, \sum_{j=1}^{p} \lambda_j = 1\}.$$

From a result of Malivert [12] we have

$$E(F,X) =$$
$$\{x \in X \mid \exists \lambda \in S_0, \sum_{i=1}^{p} \lambda_i[(B_i x + t_i)A_i - (A_i x + s_i)B_i](y - x) \geq 0 \ \forall y \in X\},$$

and

$$WE(F,X) =$$
$$\{x \in X \mid \exists \lambda \in S_1, \sum_{i=1}^{p} \lambda_i[(B_i x + t_i)A_i - (A_i x + s_i)B_i](y - x) \geq 0 \ \forall y \in X\}.$$

Define the function $g_j : S_j \times X \to R$ $(j = 0, 1)$ by setting, for each $(\lambda, x) \in S_j \times X$,

$$g_j(\lambda, x) := -\min_{y \in X} \sum_{i=1}^{p} \lambda_i[(B_i x + t_i)A_i - (A_i x + s_i)B_i]y,$$

Denote by C the $(p \times n)$- matrix whose ith row is $t_i A_i - s_i B_i$, $(i = 1, ..., p)$.

Proposition 2.1. (i) *For each j $(j = 0, 1)$, g_j is a continuous biconvex function on $S_j \times X$.*

(ii) *$g_j(\lambda, x) + \lambda C x \geq 0$ for all $(\lambda, x) \in S_j \times X$.*

(iii) *Problem* (P_j) $(j = 0, 1)$ *can be formulated as*

$$\min f(x) \tag{P_j}$$

subject to

$$x \in X, \ \lambda \in S_j, \ h_j(\lambda, x) := g_j(\lambda, x) + \lambda C x = 0.$$

Proposition 2.1 can be proved similarly as the proof of Proposition 2.1' given in [21]. For the sake of completeness we give here a proof for Proposition 2.1 that is different from that given in [21]. For the proof we need the following lemma:

Lemma 2.1. *Let A and B be two convex sets and $g_t : A \times B \to R$ $(t \in T)$ be a family of bilinear functions on $A \times B$. Then the function $g(x,y) := \sup_{t \in T} g_t(x,y)$ is biconvex on $A \times B$.*

Proof. Let $y \in B$ be fixed and $x, x' \in A$. Then for every $0 \leq \lambda \leq 1$ we have

$$\begin{aligned} g(\lambda x + (1-\lambda)x', y) &= \sup_t g_t(\lambda x + (1-\lambda)x', y) \\ &= \sup_t \{\lambda g_t(x,y) + (1-\lambda) g_t(x', y)\} \\ &\leq \lambda \sup_t g_t(x,y) + (1-\lambda) \sup_t g_t(x', y) \\ &= \lambda g(x,y) + (1-\lambda) g(x', y). \end{aligned}$$

In the same way we can prove that $g(x, .)$ is convex on B when x is fixed. ∎

Proof of Proposition 2.1. By the definition of $g_j(x,y)$ the assertion (i) is immediate from Lemma 2.1.

To prove (ii) we observe that

$$\min_{y\in X}\sum_{i=1}^{p}\lambda_i[(B_ix+t_i)A_i-(A_ix+s_i)B_i]y$$
$$\leq\sum_{i=1}^{p}\lambda_i[(B_ix+t_i)A_i-(A_ix+s_i)B_i]x\ \forall x\in X.$$

Then, since

$$\sum_{i=1}^{p}\lambda_i[(B_ix+t_i)A_i-(A_ix+s_i)B_i]x=\sum_{i=1}^{p}\lambda_i(t_iA_i-s_iB_i)x,$$

we have

$$\min_{y\in X}\sum_{i=1}^{p}\lambda_i[(B_ix+t_i)A_i-(A_ix+s_i)B_i]y\leq\sum_{i=1}^{p}\lambda_i(t_iA_i-s_iB_i)x.$$

Thus by the definitions of $g_j(\lambda,x)$ and the matrix C we have $g_j(\lambda,x)+\lambda Cx\geq 0$ for all $(\lambda,x)\in S_j\times X$.

From (ii) and the definition of $g_j(\lambda,x)$ the assertion (iii) is straightforward. ■

From Proposition 2.1 we see that Problem (P_1) of minimizing a real valued function over the weakly efficient set of Problem (VP) can be formulated in the form of Problem (P). Note that since S_0 is open, Problem (P_0) is not of the form of (P). However, in an important special case where F is linear, from a result of Philip [15], the open set S_0 appeared in Problem (P_0) can be replaced by the simplex

$$\Lambda_0:=\{\lambda\in R^p\mid\sum_{i=1}^{p}\lambda_i=1,\ \lambda_i\geq\delta,\ i=1,\ldots,p\}\tag{2}$$

with δ being a sufficiently small positive number. So, in the linear case both the problems of optimization over the efficient and weakly efficient sets take the form of (P).

2.2. Bilevel Programming

Consider the following bilevel programming problem (BL)

$$\min_{u,v}f_1(u,v):=a(u)+b(v)\tag{3}$$

subject to

$$(u,v)\in X_1:=\{(u,v)\mid A_1u+B_1v\leq p^1,\ u,v\geq 0\}\tag{4}$$

where v solves the (follower) linear program (P_u)

$$\min_{v}f_2(u,v):=cu+dv\tag{5}$$

subject to

$$(u,v) \in X_2 := \{(u,v) \mid A_2u + B_2v \le p^2, u, v \ge 0\}, \tag{6}$$

where $a(u)$ and $b(v)$ are continuous functions on R^n and R^q respectively, $c \in R^n$, $d \in R^q$, $p^1 \in R^{m_1}$, $p^2 \in R^{m_2}$ and A_i, B_i ($i = 1,2$) are suitable given matrices. Note that when $a(u)$ and $b(v)$ are linear, (BL) becomes a linear bilevel problem.

For a given u the inner problem (5)–(6) is a linear program. The dual problem of this linear program, denoted by (D_u), is

$$\max_{\lambda} \lambda(A_2u - p^2) \tag{D$_u$}$$

subject to

$$\lambda \ge 0, B_2^T\lambda \ge -d. \tag{7}$$

Suppose that for each given $u \ge 0$ the inner (follower) problem (P$_u$) has a finite optimal solution. Then, by the duality of linear programming, the dual problem (D$_u$) has also an optimal solution and their optimal values coincide. Let

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \quad B = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, \quad p = \begin{pmatrix} p^1 \\ p^2 \end{pmatrix}.$$

Then the bilevel programming problem (BL) given by (3)–(6) can be reformulated equivalently as

$$\min_{u,v} f(u,v) := a(u) + b(v) \tag{8}$$

subject to

$$Au + Bv \le p, \ u, v \ge 0, \quad B_2^T\lambda \ge -d, \ \lambda \ge 0, \tag{9}$$

$$h(\lambda, u, v) := dv - \lambda(A_2u - p^2) = 0. \tag{10}$$

From the duality of linear programming we have $h(\lambda, u, v) \ge 0$ for every (λ, u, v) satisfying (9). By setting $x := (u,v)$ and

$$X := \{x = (u,v) \mid Au + Bv \le p, \ u, v \ge 0\}, \tag{11}$$

$$\Lambda := \{\lambda \mid B_2^T\lambda \ge -d, \ \lambda \ge 0\} \tag{12}$$

we see that the problem given by (8)–(10) is of the form (P). Notice that in this case the function $h(\lambda, x)$ is bilinear on $R_+^{m_2} \times X$.

In order to compare primal and dual formulations we briefly describe here primal forms of optimization problem over the efficient set and linear bilevel programming problem. For more details about the primal forms of these problems we refer the readers to the references [2, 4, 8, 11, 22].

It is well known (see eg. [2, 4, 8]) that the problem of optimizing a function f over the efficient set of (VP) can be equivalently formulated as

$$\min\{f(x) | x \in X, r(x) = 0\},$$

where the function r is concave and given as

$$r(x) := \min\{e(F(y) - F(x)) \,|F(y) \ge F(x), y \in X\}$$

with $e \in R^p$ being the vector of components one. Note that the effective domain of r, that makes the problem nonconvex, is not given explicitly.

Similarly, for the bilevel problem (BL) we define

$$r(u) := \min\{dv | v \geq 0, A_2u + B_2v \leq p^2\},$$

and

$$h(u, v) := dv - r(u).$$

It is not difficult to check (see also [2]) that h is polyhedral concave, finite nonnegative on X_2. Moreover a point (u, v) is an optimal solution of the bilevel program (BL) if and only if it is an optimal solution of the problem

$$\min\{a(u) + b(v) | (u, v) \in X, h(u, v) = 0\}.$$

Although $h(u, v) \geq 0$ for every $(u, v) \in X$ this problem is not of the form of Problem (P) because of the joint constraint $(u, v) \in X$.

3. Penalty Function Methods

We return to Problem (P). Let $L_t(\lambda, x)$ denote the Lagrangian function with respect to the constraint $h(\lambda, x) = 0$. That is $L_t(\lambda, x) := f(x) + th(\lambda, x)$. For each $t \geq 0$ we consider the penalized problem

$$\alpha(t) := \min\{L_t(\lambda, x) | \lambda \in \Lambda, x \in X\}. \tag{P_t}$$

The following lemma is well known [7].

Lemma 3.1. (i) *$\alpha(t)$ is a nondecreasing function on R_+ and bounded from above by α_*.*

(ii) *If (λ^t, x^t) is an optimal solution of Problem (P_t) for some $t > 0$, and $x^t \in D$, then (λ^t, x^t) is an optimal solution to (P).*

For each $t \geq 0$, denote by $S(t)$ the set of global optimal solutions to (P_t). Take

$$t^* := \sup\{t \geq 0 | h(\lambda, x) > 0 \text{ for some } (\lambda, x) \in S(t)\}. \tag{13}$$

We agree to take $t^* = 0$ if the set over which the supremum takes place is empty. In what follows to a number $t > t^*$ we shall refer as an *exact penalty parameter.*

Lemma 3.2. (i) *If $0 \leq t < t^*$ then $h(\lambda, x) > 0$ for every $(\lambda, x) \in S(t)$.*

(ii) *If $t > t^*$ then $h(\lambda, x) = 0$ for every $(\lambda, x) \in S(t)$.*

Proof. Let $0 \leq t < t^*$ and $(\lambda, x) \in S(t)$. By the definition of the supremum there must exist $t < t' \leq t^*$ such that $h(\lambda', x') > 0$ for some $(\lambda', x') \in S(t')$. Since $(\lambda, x) \in S(t)$, $(\lambda', x') \in S(t')$, we have

$$f(x) + th(\lambda, x) \leq f(x') + th(\lambda', x')$$

$$f(x') + t'h(\lambda', x') \leq f(x) + t'h(\lambda, x).$$

Adding these two inequalities after a simple arrangement we have

$$(t'-t)h(\lambda',x') \le (t'-t)h(\lambda,x)$$

which together with $t'-t>0$ and $h(\lambda',x')>0$ implies $h(\lambda,x)>0$.

The assertion (ii) is immediate from the definition of t^*. ∎

Theorem 3.1. *Suppose that f is continuous on X and h is continuous on $\Lambda \times X$ and that the feasible domain of (P) is not empty. Then for every $t \ge 0$, Problem (P_t) has an optimal solution (λ^t, x^t) satisfying*

(i) *If $h(\lambda^t,x^t)=0$ then (λ^t,x^t) is an optimal solution to (P).*

(ii) *If $h(\lambda^t,x^t)>0$ for every t then any cluster point of the sequence $\{(\lambda^t,x^t)\}$ is an optimal solution to (P).*

(iii) $\lim_{t\to+\infty} f(x^t)+th(\lambda^t,x^t)=\alpha_*$.

Proof. The existence of an optimal solution of (P_t) is immediate from the compactness of $\Lambda \times X$ and the continuity of f and h. If $h(\lambda^t,x^t)=0$ then $x^t \in D$. Thus (i) follows from Lemma 3.1.

To prove (ii) let (λ^*,x^*) be any cluster point of the sequence $\{(\lambda^t,x^t)\}$. Then there exists a subsequence of $\{(\lambda^t,x^t)\}$ that, for simplicity of notation, we also denote by $\{(\lambda^t,x^t)\}$. Since (λ^t,x^t) is optimal to (P_t), we have

$$f(x^t)+th(\lambda^t,x^t) \le f(x)+th(\lambda,x)\ \forall\ \lambda\in\Lambda, x\in X.$$

If (λ,x) is feasible for (P), then $h(\lambda,x)=0$. For such a point we have

$$f(x^t)+th(\lambda^t,x^t) \le f(x)\ \forall\ x\in D$$

from which it follows that

$$0 \le h(\lambda^t,x^t) \le \frac{f(x)-f(x^j)}{t} \le \frac{2f(X)}{t}, \tag{14}$$

where $f(X) := \max_{x\in X}|f(x)| < +\infty$. Thus $h(\lambda^t,x^t)\to 0$ as $t\to+\infty$. Since h is continuous, and $(\lambda^t,x^t)\to(\lambda^*,x^*)$ we have $h(\lambda^*,x^*)=0$. On the other hand, since $h(\lambda^t,x^t)\ge 0$ we can write

$$f(x^t) \le f(x^t)+th(^t,x^t)=\alpha(t)\le \alpha_*\ \forall\ t.$$

Letting $t\to+\infty$ we obtain in the limit that $f(x^*)\le\alpha_*$ which together with $h(\lambda^*,x^*)=0$, $\lambda^*\in\Lambda$, $x^*\in X$ shows that (λ^*,x^*) is a global optimal solution to (P).

(iii) Since $\alpha(t)$ is monotone and bounded from above by α_*, we have $\lim_{t\to+\infty}\alpha(t)\le\alpha_*$.

On the other hand, since $f(x^*)=\alpha_*$, it follows that

$$\alpha_* = f(x^*) = \lim_{t\to+\infty} f(x^t) \le \lim_{t\to+\infty}\alpha(t).$$

Hence $\alpha_* = \lim_{t\to+\infty}\alpha(t)$. ∎

From Lemma 3.1 and Theorem 3.1 we see that one can solve Problem (P) by solving a sequence of linearly constrained penalized Problem (P_t) as follows:

Set $t_0 > 0$ and solve (P_{t_0}) to obtain an optimal solution (λ^0, x^0). If $h(\lambda^0, x^0) = 0$, then (λ^0, x^0) is optimal for (P). Otherwise if $h(\lambda^0, x^0) > 0$, then set $t_1 > t_0$ (for example $t_1 = 2t_0$) and solve Problem (P_{t_1}) and so on.

Notice that in view of Lemma 3.2, except for $t = t^*$, the fact that $h(\lambda, x) > 0$ or $h(\lambda, x) = 0$ does not depend on the choice of (λ, x) from the set $S(t)$.

As we have seen in the preceding section a point $x \in X$ is efficient (resp. weakly efficient) if $h_0(\lambda, x) = 0$ (resp. $h_1(\lambda, x) = 0$) for some $\lambda \in S_0$ (resp. $\lambda \in S_1$). Since h_0 and h_1 is nonnegative, we may agree to say that a point $x \in X$ is an ϵ-efficient if $h_0(\lambda, x) \leq \epsilon$ for some $\lambda \in S_0$. Similarly, a point $x \in X$ is said to be an ϵ- weakly efficient if $h_1(\lambda, x) \leq \epsilon$ for some $\lambda \in S_1$. Following this terminology we call a point $(\lambda, x) \in S \times X$ an ϵ-feasible solution to Problem (P) if $h(\lambda, x) \leq \epsilon$.

Remark 1. In the proof of Theorem 3.1 we see, from (14), that any point $(\lambda^t, x^t) \in S(t)$ is an ϵ-feasible solution to (P) whenever $t \geq \frac{2f(X)}{\epsilon}$.

In an important special case where f is concave (in particular linear) and $h(\lambda, .)$ is concave on X for each fixed $\lambda \in \Lambda$, the exact penalty is ensured by the following theorem.

Theorem 3.2. *In addition to the assumptions in Theorem 3.1 we assume that the functions $f(.)$ and $h(\lambda, .)$, for each $\lambda \in \Lambda$, are concave on X. Then t^* is finite.*

Proof. From the concavity of the objective function $L_t(\lambda, x) := f(x) + th(\lambda, x)$ on X for each $\lambda \in \Lambda$ fixed, Problem (P_t) attains its optimal value at a vertex of X. Let $V(X)$ denote the set of the vertices of X. Then

$$\alpha(t) = \min\{L_t(\lambda, v) : \lambda \in \Lambda, v \in V(X)\}.$$

We consider two cases

Case 1. $V(X) \subset D$. In this case for every $t \geq 0$ one has $S(t) \cap D \neq \emptyset$. Thus $t^* = 0$.

Case 2: $V(X) \not\subset D$. Then there is $v \in V(X)$ such that $h(\lambda, v) > 0$ for every $\lambda \in \Lambda$. Set

$$M_0 := \min\{h(\lambda, v) | v \in V(X), h(\lambda, v) > 0 \ \forall \ \lambda \in \Lambda\}.$$

Since Λ is compact, $h(., v)$ is continuous on Λ while $V(X)$ is finite, we see that $M_0 > 0$.

Let $(\lambda^t, x^t) \in S(t)$ and (λ^0, x^0) such that $h(\lambda^0, x^0) = 0$. Then

$$f(x^t) + th(\lambda^t, x^t) \leq f(x^0) + th(\lambda^0, x^0) = f(x^0).$$

If $h(\lambda^t, x^t) > 0$ then

$$t \leq \frac{f(x^0) - f(x^t)}{h(\lambda^t, x^t)} \leq 2\frac{f(X)}{M_0} < +\infty.$$

Consequently we conclude that $h(\lambda^t, x^t) = 0$ for every $(\lambda^t, x^t) \in S(t)$ whenever $t > 2f(X)/M_0$ which implies that $t^* < +\infty$. ■

Note that in view of Lemma 3.2 every optimal solution to (P_t) is also an optimal solution to Problem (P) provided $t > t^*$.

Remark 2. From the result of the previous section, Problems (0) and (1) of minimizing the function f over the efficient and weakly efficient sets of the multiple objective linear program

$$\text{vmin}\{Cx|\ x \in X\}$$

can be formulated as

$$\min\{f(x)\ |x \in X, \lambda \in \Lambda_0, h(\lambda, x) := g(\lambda) + \lambda Cx \leq 0\} \qquad (0')$$

and

$$\min\{f(x) : x \in X, \lambda \in \Lambda_1, h(\lambda, x) := g(\lambda) + \lambda Cx \leq 0\} \qquad (1')$$

respectively, where $g(\lambda) = \max_{x \in X} \lambda Cx$ is convex (independent of x). So when f is concave on X, these problems satisfy the assumptions of Theorem 3.2. Likewise for the bilevel programming problem given by (8)–(10) by setting $x = (u, v)$ one can see that when $a(u)$ and $b(v)$ are concave, the assumptions of Theorem 3.2. are also fulfilled, since the objective function is concave and $h(\lambda, x)$ is bilinear. Thus, for both these problems we have the exact penalty, i.e., t^* is finite.

4. Application to Linear Bilevel Programming

In this section we apply the results obtained in the preceding sections to solving the problem given by (8)–(10). As we have seen this problem is the dual form of the bilevel problem (BL). By setting $x = (u, v)$, it can be rewritten as

$$\min f(x) := a(u) + b(v) \qquad \text{(DBL)}$$

subject to

$$x \in X := \{x = (u, v)|Au + Bv \leq p,\ u, v \geq 0\},$$

$$\lambda \in \Lambda := \{\lambda \in R^{m_2}|B_2^T\lambda \geq -d, \lambda \geq 0\},$$

$$h(\lambda, x) := h(\lambda, u, v) = dv - \lambda(A_2u - p^2) = 0.$$

Under the assumptions of Theorem 3.2 this problem has finite exact penalty parameters, i.e., $0 \leq t^* < +\infty$, such that for every $t > t^*$, it and the penalized problem

$$\min\{L_t(\lambda, x) := f(x) + th(\lambda, x)\ |\ \lambda \in \Lambda, x = (u, v) \in X\}, \qquad (\text{DBL}_t)$$

have the same solution-set. Since $f(x)$ is linear on X and $h(\lambda, x)$ is bilinear on $R_+^{m_2} \times X$, this problem is a linearly constrained bilinear program that could be solved by some existing methods (see e.g. [9, 10, 17, 18] and references therein). Note that although the existence of an exact penalty parameter is ensured, determining it is difficult except for some special cases. Usually one takes $t > 0$ large enough. However from computational experiences on penalty function methods, it is well recognized that with t too large the penalized problem would be instable whereas with t small, solutions of the penalized problem may not be optimal for the original problem.

In a special case, when the objective function of the leader depends on the objective function of the follower, the exact penalty parameter t^* can be estimated. Namely we have the following proposition.

Proposition 4.1. *Suppose that the polyhedron X is bounded and $b(v) = \theta(dv)$ where θ is a continuously differentiable function on an open interval containing $[m_0, m_1]$, where*

$$m_0 := \min\{dv : A_2u + B_2v \le p^2, u, v \ge 0\},$$

$$m_1 := \max\{dv : A_2u + B_2v \le p^2, u, v \ge 0\}.$$

Let

$$\theta'_* := \max\{\theta'(\tau) : \tau \in [m_0, m_1]\}.$$

Then

$$t^* \le \bar{t} := \max\{0, -\theta'_*\}.$$

Proof. Let $t' > 0$ be arbitrary and $t = t' + \bar{t}$. Consider Problem (DBL_t)

$$\min L_t(\lambda, u, v) := a(u) + \theta(dv) + (t' + t^*)[dv - \lambda(A_2u - p^2)]$$

subject to

$$(u, v) \in X,\ B_2^T\lambda \ge -d, \lambda \ge 0,\ \ dv - \lambda(A_2u - p^2) \le 0.$$

Let (λ^t, u^t, v^t) be an optimal solution of this penalized problem. Then

$$\begin{aligned} &a(u^t) + \theta(dv^t) + (t' + \bar{t})[dv^t - \lambda^t(A_2u^t - p^2)] \\ &\le a(u) + \theta(dv) + (t' + \bar{t})[dv - \lambda(A_2u - p^2)] \end{aligned} \tag{15}$$

for every point (λ, u, v) feasible for (DBL_t).

Let $\bar{v}^t$ be an optimal solution of the linear program (P_{u^t}) and $\bar{\lambda}^t$ be that of the dual program of (P_{u^t}). Applying (15) with $(\lambda, u, v) = (\bar{\lambda}^t, u^t, \bar{v}^t)$ and observing

$$d\bar{v}^t - \bar{\lambda}^t(A_2u^t - p^2) = 0,$$

we obtain

$$\begin{aligned} t'[dv^t - \lambda^t(A_2u^t - p^2)] \le\ &\theta(d\bar{v}^t) - \theta(dv^t) + \bar{t}(d\bar{v}^t - dv^t) \\ &+ \bar{t}[\lambda^t(A_2u^t - p^2) - \bar{\lambda}^t(A_2u^t - p^2)] \end{aligned} \tag{16}$$

Using the mean-value theorem for θ we have

$$\theta(d\bar{v}^t) - \theta(dv^t) = \theta'(\tau_0)(d\bar{v}^t - dv^t)$$

for some $\tau_0 \in [d\bar{v}^t, dv^t]$. This and (16) imply

$$t'[dv^t - \lambda^t(A_2u^t - p^2)] \leq [\bar{t} + \theta'(\tau_0)](d\bar{v}^t - dv^t) + \bar{t}[\lambda^t(A_2u^t - p^2) - \bar{\lambda}^t(A_2u^t - p^2)] \quad (17)$$

Since $\bar{v}^t$ is optimal to Problem (P_{u^t}) and $\bar{t} = \max\{0, -\theta'_*\}$, we have

$$(d\bar{v}^t - dv^t)[\bar{t} + \theta'(\tau_0)] \leq 0. \quad (18)$$

On the other hand, since $\bar{\lambda}^t$ is optimal for the dual problem of (P_{u^t}), we have

$$\bar{t}[\lambda^t(A_2u^t - p^2) - \bar{\lambda}^t(A_2u^t - p^2)] \leq 0. \quad (19)$$

Thus, the right-hand side of (17) is nonpositive. Using again (17) we get

$$t'[dv^t - \lambda^t(A_2u^t - p^2)] = 0$$

which, since $t' > 0$, implies

$$h(\lambda^t, u^t, v^t) = dv^t - \lambda^t(A_2u^t - p^2) = 0.$$

Hence (λ^t, u^t, v^t) solves Problem (DBL). Since $t = t' + \bar{t}$ with $t' > 0$ arbitrary, we deduce that $t^* \leq \max\{0, -\theta'_*\}$ that proves the proposition. ∎

Remark 3. When $\theta(t)$ is affine, i.e., $\theta(t) = \xi t + \xi_0$ then $\theta'(t) = \xi$. In this case $t^* = 0$ if $\xi \geq 0$ and $t^* \leq -\xi$ if $\xi < 0$.

For each $t \geq 0$ we define the function $\phi_t(\lambda)$ by setting

$$\phi_t(\lambda) := \min\{L_t(\lambda, x) := f(x) + th(\lambda, x) | x \in X\}. \quad (20)$$

Clearly

$$\alpha(t) = \min\{\phi_t(\lambda) | \lambda \in \Lambda\}.$$

Moreover if λ^t is an optimal solution of this problem and x^t is an optimal solution of problem (20) with $\lambda = \lambda^t$, then (λ^t, x^t) is an optimal solution to (DBL_t).

The algorithm to be described below is a decomposition branch-and-bound procedure using the convex envelope functions for bounding and an adaptive simplicial subdivision on Λ for branching. First we describe these two operations.

4.1. Bounding by the Convex Envelope Function

It is well known that [6, 9] the convex envelope function of a concave function $f(x)$ on an m_2-simplex S is an affine function of the form $\varphi_S(\lambda) := \langle l, \lambda \rangle + \xi$ where $l \in R^{m_2}$ and $\xi \in R$ are uniquely determined by the system of linear equations

$$\langle l, v^i \rangle + \xi = f(v^i) \ (i = 0, 1, ..., m_2)$$

with v^i $(i = 0, ...m_2)$ being the vertices of S. Note that at the vertices of the simplex the values of a concave function and of its convex envelope function coincide [6, 9].

We shall use the convex envelope function of the function $\phi_t(\lambda)$ for computing lower bounds. Namely, let S be an m_2-dimensional simplex vertexed at $v^0, \dots, v^{m_2}$. Let $\varphi_{tS}(\lambda)$ denote the convex envelope function of ϕ_t on S. Take

$$\beta(S) := \min\{\varphi_{tS}(\lambda) | \lambda \in S \cap \Lambda\}.$$

Then $\beta(S) \leq \min\{\phi_t(\lambda) | \lambda \in S \cap \Lambda\}$.

Since $\lambda \in \Lambda$, expressing

$$\lambda = \sum_{i=0}^{m_2} \xi_i v^i, \xi_i \geq 0, \sum_{i=0}^{m_2} \xi_i = 1,$$

we have

$$\beta(S) = \min \varphi_{tS}(\xi) \tag{LS}$$

subject to

$$\xi = (\xi_0, \dots, \xi_{m_2}) \geq 0, \sum_{i=0}^{m_2} \xi_i = 1, \sum_{i=0}^{m_2} \xi_i B_2^T v^i \geq -d.$$

4.2. An Adaptive Simplicial Subdivision

Simplicial subdivisions are widely used in global optimization. For the lower bounding using the convex envelope function defined above, we shall determine a simplicial subdivision as follows.

Let $\lambda^S \in S$ be an optimal solution for the program defining $\beta(S)$. If $\lambda^S \in V(S)$, then $\phi_t(\lambda^S) = \varphi_{tS}(\lambda^S)$. In this case the lower bound $\beta(S)$ is the exact bound, and therefore the simplex S is not of interest in further consideration. So we assume $\lambda^S \notin V(S)$. Let

$$\lambda^S = \sum_{i=0}^{m_2} \xi_i v^i$$

with $\xi_i \geq 0$, $\sum_{i=0}^{m_2} \xi_i = 1$. Since $\lambda^S \notin V(S)$, the index-set

$$I(\lambda^S) := \{i : \xi_i > 0\} \tag{21}$$

has at least two elements. We then subdivide S into subsimplices S_i ($i \in I(\lambda(S))$), where each S_i is obtained from S by replacing the vertex v^i of S by λ^S. We call λ^S the *subdivision point*.

In comparison with the bisection via the midpoint of a longest edge, this subdivision has the advantage that it takes the information obtained from the bounding operation into account. The disadvantage however is that it in general does not ensure convergence. This suggests combining these two subdivisions. In the algorithm to be described below we shall use the following rule.

Rule 1. [20] Suppose that S is the simplex to be subdivided at iteration k. Let $v^0, \dots, v^{m_2}$ be the vertices of S, and $\lambda^S \in S \setminus V(S)$. Let N be a given arbitrary natural number. Then we subdivide S by the above simplicial subdivision, where the subdivision point is the midpoint of a longest edge of S if k is a multiple of N. Otherwise it is λ^S.

As usual for a given $\epsilon \geq 0$ we call a point z^* an ϵ- *optimal solution* to the problem

$$f_* = \min\{f(z) : z \in Z\},$$

if

$$z^* \in Z, \text{ and } f(z^*) - f_* \leq \epsilon(|f(z^*)| + 1).$$

Now we describe steps of an algorithm for solving problem (DBL) by using the penalized problem (DBL_t). The algorithm to be described below will determine an exact penalty parameter iteratively as follows:

With a penalty parameter $t > 0$ given in advance the algorithm computes an ϵ-optimal solution to Problem (DBL_t). If the obtained solution is feasible for (DBL), then the algorithm terminates yielding an ϵ-optimal solution to Problem (DBL). Otherwise, t increases by $\Delta > 0$, the objective function thereby is changed. Then the algorithm recomputes lower bounds for the new objective function on each generated partition simplex, and thereby upper bounds are improved. The penalty parameter is also increases iteratively in such a way (cases 2a and 2b of Step 2) that the obtained upper bounds tend to the optimal value. Since the exact parameter is assumed, the steps involving updating penalty parameter (cases 1b and 2b) cannot occur infinitely many times. In order to save computational costs, the algorithm verifies feasibility for each newly generated simplex S by computing $\gamma(S)$ (step 4). The algorithm deletes every generated simplex S if it is infeasible, i.e. $\gamma(S) > 0$.

The algorithm now can be described in detail as follows.

ALGORITHM 1 (no $t > t^*$ is known in advance)

Initialization. Take $\epsilon \geq 0$, $t > 0$, $\Delta > 0$ and a natural number N. Construct an m_2- simplex $S_0 \supset \Lambda$. Let $\Gamma_0 := \{S_0\}$ and take

$$\alpha_{-1} = \begin{cases} +\infty, & \text{if no feasible point is available,} \\ L_t(\lambda^{-1}, x^{-1}), & \text{otherwise,} \end{cases}$$

where (λ^{-1}, x^{-1}) is the best currently known feasible point.

Let $k \leftarrow 0$.

Iteration k ($k = 0, 1....$).

For each $S \in \Gamma_k$ solve the linear program

$$\beta(S) = \min\{\varphi_{tS}(\lambda) : \lambda \in S \cap \Lambda\}$$

to obtain an optimal solution λ^S.

For each obtained λ^S solve the linear program

$$\alpha(S) := \min\{L_t(\lambda^S, x) : x \in X\}$$

to obtain a basic solution $x^S \in V(X)$.

Step 0. Update the currently best lower and upper bounds by setting

$$\beta_k := \min\{\beta(S) | S \in \Gamma_k\}.$$

$$\alpha_k := \min\{\alpha_{k-1}, L_t(\lambda^S, x^S) : S \in \Gamma_k\}.$$

Let

$$(\lambda^k, x^k) \in \{(\lambda^{k-1}, x^{k-1}), (\lambda^S, x^S) : S \in \Gamma_k\}$$

such that

$$L_t(\lambda^k, x^k) = \alpha_k.$$

Step 1. If

$$\alpha_k - \beta_k \leq \epsilon(|\alpha_k| + 1)$$

we distinguish two cases

1a. If $h(\lambda^k, x^k) \leq 0$, then terminate: (λ^k, x^k) is an ϵ-optimal solution to (DBL$_t$).

1b. If $h(\lambda^k, x^k) > 0$, then set $t \leftarrow t + \Delta$ and go to iteration k (with k unchanged).

Step 2. If

$$\alpha_k - \beta_k > \epsilon(|\alpha_k| + 1)$$

select $S_k \in \Gamma_k$ such that

$$\beta(S_k) = \beta_k := \min\{\beta(S) | S \in \Gamma_k\}.$$

Consider two cases:

2a. If

$$h(\lambda^k, x^k) \leq \alpha_k - \beta_k,$$

then go to Step 3.

2b. If

$$h(\lambda^k, x^k) > \alpha_k - \beta_k$$

then set $t \leftarrow t + \Delta$ and go to iteration k (with k unchanged).

Step 3. Subdivide S_k by a w- radial subdivision according to Rule 1, i.e. we take the subdivision point $w = \lambda^k$ if k is not a multiple of N, and $w = (v^i + v^j)/2$ otherwise, where v^i, v^j are the two end points of a longest edge of S.

Step 4. For each $i \in I(\lambda^k)$ defined by (21) with $\lambda^S = \lambda^k$ determine

$$\gamma(S_{ki}) := \min_{v^i \in V(S_{ki})} \min\{h(v^i, x) | x \in X\}.$$

Let

$$I_0(\lambda^k) := \{i \in I(\lambda^k) \,|\, \gamma(S_{ki}) \leq 0\}.$$

Step 5. For each $i \in I_0(\lambda^k)$ compute

$$\beta(S_{ki}) := \min\{\varphi_{tS_{ki}}(\lambda) \mid \lambda \in \Lambda \cap S_{ki}\} = \varphi_{tS_{ki}}(\lambda^{ki}).$$

Set

$$\alpha(S_{ki}) = L_t(\lambda^{ki}, x^{ki}) := \min\{L_t(\lambda^{ki}, x) \mid x \in X\}$$

where, as before, $\lambda^{ki} \in S_{ki}$ is an optimal solution of the problem determining $\beta(S_{ki})$.

Define

$$\Gamma_{k+1} := (\Gamma_k \setminus \{S_k\}) \cup \{S_{ki} : i \in I_0(\lambda^k)\}$$

and update the upper and lower bounds by setting

$$\alpha_{k+1} := \min\{\alpha_k, \alpha(S_{ki}) : i \in I_0(\lambda^k)\}, \ \beta_{k+1} := \min\{\beta(S) \mid S \in \Gamma_{k+1}\}.$$

Let $k \leftarrow k+1$ and go to Step 1 of Iteration k.

Before proving the convergence we have some remarks on the algorithm.

Remark 4. When $a(u)$ and $b(v)$ are linear, the function $L_t(\lambda, x)$ is bilinear on $\Lambda \times X$. Thus, the algorithm for this case involves only linear programs.

Remark 5. Since $h(\lambda, x)$ is bilinear, from the definition of $\gamma(S)$, it is clear that $\gamma(S) > 0$ if and only if $h(\lambda, x) > 0$ for all $(\lambda, x) \in S \times X$. This means that $S \times X$ does not contain a feasible point of (DBL).

Theorm 4.1. (i) *If the algorithm terminates at iteration k, then (λ^k, x^k) is an ϵ-optimal solution to* (DBL).

(ii) *The algorithm does not terminate only when $\epsilon = 0$. In this case $\alpha_k \searrow \alpha_*$ and $\beta_k \nearrow \alpha_*$ as $k \to +\infty$. Moreover, any cluster point of the sequence $\{(\lambda^k, x^k)\}$ is a global optimal solution to* (DBL).

Proof. (i) If the algorithm terminates at some iteration k, then

$$\alpha_k - \beta_k \leq \epsilon(|\alpha_k| + 1)$$

and $h(\lambda^k, x^k) \leq 0$. Since β_k is a lower bound, α_k is an upper bound and

$$\alpha_k = f(x^k) + th(\lambda^k, x^k) = f(x^k),$$

it follows that $f(x^k) - \beta_k \leq \epsilon(|f(x^k)| + 1)$. Hence (λ^k, x^k) is an ϵ-optimal solution to (DBL).

(ii) Suppose now that the algorithm does not terminate. First we show that case 1b of step 1 cannot occur infinitely many times. Indeed once case 1b occurs, the penalty parameter increases by Δ. Since $\Delta > 0$, at some iteration k the penalty parameter t must exceed t^*. Then at Step 1 we have $h(\lambda^k, x^k) = 0$, and therefore case 1b cannot happen infinitely many times.

Similarly case 2b cannot also happen infinitely many times. In fact at case 2b

$$h(\lambda^k, x^k) > \alpha_k - \beta_k > 0.$$

Since $\alpha_k = f(x^k) + th(\lambda^k, x^k)$, we have

$$(t-1)h(\lambda^k, x^k) < \beta_k - f(x^k). \tag{22}$$

Observing that $x^k \in V(X)$ we obtain

$$h(\lambda^k, x^k) \geq \min_{\lambda \in \Lambda}\{h(\lambda, x) | x \in V(X), h(\lambda, x) > 0\} = \delta > 0. \tag{23}$$

Noting that the sequences $\{\beta_k\}$ and $\{f(x^k)\}$ are bounded we see from (22) and (23) that t cannot go to $+\infty$. Thus case 2b cannot happen infinitely many times.

Consequently if the algorithm does not terminate, then it generates a nested sequence of partition simplices. For simplicity of notation we also denote this sequence by $\{S_k\}$. From Rule 1 for subdivision, by Theorem 2 [20], there exists a subsequence $\{S_{kj}\}$ of the partition simplices $\{S_k\}$ satisfying

$$w^{k_j} = \lambda^{k_j} \quad \forall j, \tag{24}$$

$$\lambda^{k_j} \to \lambda^*, \; d(\lambda^*, V(S_{k_j})) \to 0 \text{ as } j \to +\infty, \tag{25}$$

where $d(\lambda^*, V(S_{kj}))$ denotes the Euclidean distance from λ^* to $V(S_{kj})$. By the rule for computing lower bounds we have

$$\beta_{k_j} = \varphi_{tS_{k_j}}(\lambda^{k_j}) \; \forall j. \tag{26}$$

Noting that, for every vertex v^{k_j} of the simplex S_{k_j}, since $\varphi_{tS_{k_j}}(\lambda)$ is the convex envelope function of $\phi_t(\lambda)$ on S_{kj}, we have

$$\phi_t(v^{k_j}) = \varphi_{tS_{k_j}}(v^{k_j}) \; \forall j. \tag{27}$$

Letting $j \to +\infty$ we obtain from (24), (25), (26), (27) that $\beta_{k_j} \to \phi_t(\lambda^*)$ as $j \to +\infty$. Since β_{k_j} is a lower bound for the optimal value of Problem (DBL$_t$) and $\lambda^* \in \Lambda$, it follows that λ^* is a minimal point of $\phi_t(\lambda)$ over Λ. Thus,

$$\lim_k \beta_k = \phi_t(\lambda^*) = \min\{f(x) + th(\lambda^*, x) : x \in X\} = f(x^*) + th(\lambda^*, x^*)$$

which implies that (λ^*, x^*) is an optimal solution to (DBL$_t$).

On the other hand, from the rule for computing upper bounds we have

$$\alpha_{k_j} = f(x^{k_j}) + th(\lambda^{k_j}, x^{k_j}) \leq f(x^*) + th(\lambda^{k_j}, x^*).$$

Letting $j \to +\infty$ and observing that the sequence $\{\alpha_k\}$ is monotone we obtain in the limit that

$$\lim_k \alpha_k \leq f(x^*) + th(\lambda^*, x^*). \tag{28}$$

Since α_k is an upper bound for the optimal value of (DBL$_t$) it follows from (28) that

$$\lim_k \alpha_k = f(x^*) + th(\lambda^*, x^*).$$

So

$$\lim_k \alpha_k = \lim_k \beta_k = f(x^*) + th(\lambda^*, x^*).$$

As the case 2b cannot happen infinitely many times, without loss of generality, we may assume that case 2a occurs for every k. Then

$$h(\lambda^k, x^k) \le \alpha_k - \beta_k \ \ \forall k.$$

From this and $h(\lambda^k, x^k) \ge 0$ for every k, letting $k \to +\infty$ we get $h(\lambda^*, x^*) = 0$. Since (λ^*, x^*) is a global optimal solution to (DBL$_t$), in view of (i) in Theorem 3.1, we deduce that (λ^*, x^*) solves (DBL) globally as well.

Now let (λ, x) be any cluster point of the sequence $\{(\lambda^k, x^k)\}$. For simplicity we assume that

$$(\lambda^k, x^k) \to (\lambda, x) \ \text{ as } k \to +\infty.$$

Since $\alpha_k = f(x^k) + th(\lambda^k, x^k)\ \forall k$ and $(\lambda^k, x^k) \in \Lambda \times X$, we obtain in the limit that

$$(\lambda, x) \in \Lambda \times X, \ \text{ and } \ \lim_k \alpha_k = \alpha_* = f(x) + th(\lambda, x)$$

which together with $\alpha_* = f(x^*) + th(\lambda^*, x^*) = f(x^*)$ imply that (λ, x) is a global optimal solution to (DBL). ■

Remark 6. In the case an exact penalty parameter is known in advance, Algorithm 1 becomes much simpler. Namely, in this case the algorithm can be described simply as follows.

ALGORITHM 2 (case $t > t^*$ is known in advance)

Initialization. Take $\epsilon \ge 0$ and a natural number N. Construct an m_2-simplex $S_0 \supset \Lambda$.

Step 0. Compute

$$\beta(S_0) := \min\{\varphi_{tS_0}(\lambda) \mid \lambda \in \Lambda\}, \tag{LP$_0$}$$

where $\varphi_{tS_0}(\lambda)$ is the convex envelope function of $\phi_t(\lambda)$ on the simplex S_0. Let

$$\beta_0 := \beta(S_0) = \varphi_{tS_0}(\lambda^0)$$

$$\alpha_0 := \min\{L_t(\lambda^0, x) | x \in X\} = L_t(\lambda^0, x^0),$$

where λ^0 is an optimal solution to Problem (LP$_0$).

Set

$$\Gamma_0 = \begin{cases} \emptyset, & \text{when } \alpha_0 - \beta_0 \le \epsilon(|\alpha_0| + 1), \\ \{S_0\} & \text{otherwise.} \end{cases}$$

Let $k \leftarrow 0$.

Iteration k $(k = 0, 1.....)$.

Step 1. If $\Gamma_k = \emptyset$, then terminate: (λ^k, x^k) is an ϵ-optimal solution to (DBL).

Step 2. If $\Gamma_k \ne \emptyset$ choose $S_k \in \Gamma_k$ such that

$$\beta(S_k) = \beta_k = \min\{\beta(S) | S \in \Gamma_k\}.$$

Step 3. Subdivide S_k by a w-radial subdivision according to Rule 1.

Step 4. For each $i \in I(\lambda^k)$ determine

$$\gamma(S_{ki}) := \min_{v^i \in V(S_{ki})} \min\{h(v^i, x) | x \in X\}.$$

Let

$$I_0(\lambda^k) := \{i \in I(\lambda^k) \mid \gamma(S_{ki}) \leq 0\}.$$

Step 5. For each $i \in I_0(\lambda^k)$ compute

$$\beta(S_{ki}) := \min\{\varphi_{tS_{ki}}(\lambda) | \lambda \in \Lambda \cap S_{ki}\},$$

$$\alpha(S_{ki}) = L_t(\lambda^{ki}, x^{ki}) := \min\{L_t(\lambda^{ki}, x) | x \in X\}.$$

Update the upper bound by setting

$$\alpha_{k+1} := \min\{\alpha_k \, | \alpha(S_{ki}) \; i \in I_0(\lambda^k)\}$$

Set

$$\Gamma_{k+1} := (\Gamma_k \setminus \{S_k\}) \cup \{S_{ki} \,|\, \alpha_{k+1} - \beta(S_{ki}) > \epsilon(|\alpha_{k+1}| + 1), \; i \in I_0(\lambda^k)\}.$$

Set $k \leftarrow k+1$ and go to Iteration k.

Comment. The global search in the above algorithms takes place in the λ-space whose dimension is m_2. So it is expected that the algorithms are efficient when m_2 is relatively small.

4.3. Computational Experiences and Results

We illustrate the algorithm by the following simple example:

$$\min\{-2u_1 + u_2 + 0.5v_1 \;\; | \, u_1 + u_2 \leq 2, u_1, u_2 \geq 0\}$$

where $v = (v_1, v_2, v_3)$ solves the program

$$\min\{-4v_1 + v_2 + 5v_3 \,|\, -2u_1 + v_1 - v_2 \leq -2.5, \; u_1 - 3u_2 + v_2 - v_3 \leq 2, \; v_1, v_2, v_3 \geq 0\}.$$

We choose $t = 5$ and solve this problem by Algorithm 2 with the initial partion simplex vertexed at $(0,0)$, $(11,0)$, $(0,11)$. At iteration 4

$$\Gamma_4 = \{S_4 := \text{conv}\,((0,0), (6,5), (0,11))\},$$

and the best upper bound $\alpha_4 = -3.25$ corresponding to the feasible point $u^* = (2,0)$, $v^* = (1.5, 0, 0)$ (this point has been found at iteration 1). The algorithms divides S_4 into two simplices

$$S_{4,1} = \text{conv}\,((4, 3.33), (6,5), (0,11)), \;\; S_{4,2} = \text{conv}\,((0,0), (4, 3.33), (0,11)).$$

The lower bounds for these simplices are $\beta(S_{4,1}) = \beta(S_{4,2}) = -3.25$. Since lower and upper bounds coincide, the algorithm terminates showing that $u^* = (2,0)$, $v^* = (1.5,0,0)$ is a global optimal solution.

In order to obtain a preliminary evaluation of the performance of the proposed algorithms, we have written computer code by PASCAL TURBO 7.0 that implements the algorithms. The code used the ordinary simplex method for solving the linear programs called for by the algorithm. To test the code we use it to solve hundreds randomly generated problems on a Pentium II personal computer. For all tested problems we take $\epsilon = 10^{-4}$. The computed results are reported in Table 1. In the table we use the following headings:

- n, p: the number of variables u and v respectively,
- m_1, m_2 : the numbers of constraints (without nonnegative one) of the leader and follower problems respectively,
- *ite* : the average number of the iterations
- s : the average number of all generated simplices
- *re.s*: the average number of the simplices stored in the memory,
- *time*: the average CPU time (in second).

Table 1

Prob.	m_1	m_2	n	p	ite	s	$re.s$	time s
1	10	3	40	10	18	28	8	20.51
2	10	3	50	10	18	27	7	27.20
3	20	3	40	10	10	17	5	31.02
4	20	3	50	10	13	20	7	61.01
5	10	4	40	10	43	78	14	59.38
6	10	4	50	10	33	61	17	57.53
7	20	4	40	10	22	41	10	87.56
8	20	4	50	10	38	72	14	143.48
9	10	5	40	10	68	117	55	107.80
10	10	5	50	10	45	98	35	89.04
11	20	5	40	10	69	154	58	241.96
12	20	5	50	10	57	117	24	376.60
13	10	6	40	10	46	79	23	106.01
14	10	6	50	10	49	124	56	129.80
15	20	6	40	10	62	114	22	262.85
16	20	6	50	10	63	116	65	279.87
17	10	7	40	10	66	208	145	155.29
18	20	7	40	10	60	178	80	389.55
19	10	7	50	10	82	211	150	226.99

The results in the table show that the algorithm could be used for solving linear bilevel programs with moderate size on a Pentium II. It appears that the running time is much more sensitive to the growth in the number m_2 of constraints of the follower problem than to the growth in the number of variables or constraints of the leader problem. The required memory however increases slowly as the program runs, since a large percentage of the generated simplices is eliminated from further consideration. A main property of this algorithm is that a feasible point found at some iteration is recognized as a global optimal solution very late.

References

1. Le T. H. An, Pham D. Tao, and Le D. Muu, Numerical solution for optimization over the efficient set by d.c. optimization algorithm, *Operations Research Letters* **19** (1996) 117–128.
2. Le T. H. An, Pham D. Tao, and Le D. Muu, Exact penalty in d.c. programming, *Vietnam J. Math.* **27** (1999) 161–178.
3. J. F. Bard, An efficient algorithm for a linear two-stage optimization problem, *Operations Research*, July-August (1983) 670–684.
4. H. P. Benson, Optimization over the efficient set, *J. Math. Anal. Appl.* **98** (1984) 562–580.
5. S. Bolintineanu and M. E. Mari, Penalisation dans l'optimisation sur l'ensemble faiblement efficient, *Recherche Operationnelle* **31** (1997) 295–310.
6. J. Falk, Lagrange multipliers and nonlinear programming, *J. Anal. Appl.* **19** (1967) 141–159.
7. A. Fiacco and G. McCormik, *Nonlinear Programming Sequential Unconstrained Minimization Techniques*, John Willey & Sons, New York, 1968.
8. J. Fulop, Lagrangian duality of concave minimization subject to linear constraints and an additional facial reverse convex constraint, *J. Optim. Theory Appl.* **81** (1996) 617–641.
9. R. Horst and H. Tuy, *Global Optimization (Deterministic Approach)*, Third Edition, Springer-Verlag, Berlin - New York, 1996.
10. H. Konno, A cutting-plan algorithm for solving bilinear programs, *Math. Program.* **11** (1976) 14–27.
11. Le T. Luc and Le D. Muu, A global optimization approach to optimization over the efficient set, *Lecture Notes in Economics and Mathematical Systems* **452** (1997) 183–195.
12. C. Malivert, Multicriteria fractional optimization, *Proceedings of the 2nd Catalans Days on Applied Mathematics*, Presses Universitaires de Perpignan, (1995) 189–198.
13. A. Migdalas, P. Pardalos, and P. Varbrand (eds.), *Multilevel Optimization: Algorithms and Applications*, Kluwer Academic Publishers, Dordrecht - Boston - London, 1998.
14. Le D. Muu and W. Oettli, An algorithm for indefinite quadratic programming with convex constraints, *Operations Research Letters* **10** (1991) 323–327.
15. J. Philip, Algorithms for the vector maximization problem, *Math. Program.* **2** (1972) 207–229.
16. R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation and Application*, John Willey & Sons, New York, 1986.
17. T. V. Thieu, Relationship between bilinear programming and concave programming, *Acta Math. Vietnam.* **2** (1980) 106–113.

18. T. V. Thieu, A note on the solution of bilinear programming problem by reduction to concave minimization, *Math. Program.* **41** (1988) 249–260.
19. P. T. Thach, H. Konno, and D. Yokota, Dual approach to optimization on the set of pareto-optimal solutions, *J. Optim. Theory and Appl.* **88** (1996) 689–707.
20. H. Tuy, Effect subdivision strategy on convergence and efficiency of some global optimization algorithms, *J. Glob. Optim.* **1** (1992) 23–33.
21. H. Q. Tuyen and Le D. Muu, Biconvex programming approach to optimization over the weakly efficient set of a multiple objective affine fractional problem, *Operations Research Letters* **28** (2001) 81–92.
22. D. J. White and G. Anandalingam, A penalty function approach for solving bi-level linear program, *J. Glob. Optim.* **3** (1993) 397–419.