ACTA MATHEMATICA VIETNAMICA
Volume 26, Number 3, 2001, pp. 333-347

METHODS FOR FINDING GLOBAL OPTIMAL SOLUTIONS TO LINEAR PROGRAMS WITH EQUILIBRIUM CONSTRAINTS

LE DUNG MUU AND NGUYEN VAN QUY

Dedicated to Pham Huu Sach on the occasion of his sixtieth birthday

Abstract. A mathematical program with equilibrium constraints is an optimization problem with two sets of variables x and y, in which some or all of its constraints are defined by a parametric variational inequality or complementarity system with y as its primary variables and x the parameter vector. This problem even in the linear case is a difficult global optimization problem because of its nested structure. We use a dual formulation of this problem to develop two decomposition branch-and-bound algorithms for finding a global optimal solution of a linear mathematical program with equilibrium constraints. The first algorithm uses a simplicial subdivision whereas the second one uses a binary tree defined by using the sign (zero or positive) of the dual variables. The searching trees in both algorithms are created in the dual space. Preliminary computational experiences and results show that on a PC computer the algorithm using binary tree can solve linear problems with equilibrium constraints up to twenty-five dual variables; the number of the primal variables may be larger.

1. Introduction

A mathematical program with equilibrium constraints, shortly MPEC, is an optimization problem with two sets of variables, $x \in R^n$ and $y \in R^m$, in which some or all of its constraints are defined by a parametric variational inequality or complementarity system with y as its primary variables and x the parameter vector. More specifically, this problem is defined as follows.

Suppose that $f : R^{n+m} \to R$, $F : R^{n+m} \to R^m$ are given functions, $D \subseteq R^{n+m}$ is a nonempty closed set, and $C : R^n \to R^m$ is a set-valued map with closed convex values, i.e, for each $x \in R^n$, $C(x)$ is a (possibly empty) closed convex subset of R^m. The set of all vectors $x \in R^n$ for which $C(x) \neq \emptyset$ is the domain of C and denoted by $\text{dom}(C)$.

Received February 26, 2001.
1991 *Mathematics Subject Classification.* 90 C29.
Key words and phrases. Equilibrium constraint, dual formulation, complementarity condition, simplicial and binary tree branching, global optimization, computational report.
This work was supported in part by the National Basic Research Program in Natural Science, Vietnam.

The function f is the overall objective function to be minimized; F is the equilibrium function of the inner problem, D is a joint upper-level feasible region of the pair (x, y), and $C(x)$ defines the restriction of the variable y for each given x. With this setup, the problem can be given as:

$$\min \ f(x, y) \tag{1}$$

$$\text{subject to } (x, y) \in D, \tag{P}$$

where y solves the parametric variational inequality

$$\text{find } \ y \in C(x): \ (v - y)^T F(x, y) \geq 0, \ \text{for all } v \in C(x) \tag{VI(x)}$$

This problem has some important applications, for example, in the design of transportation networks, in economic modeling. The MPEC problem includes, as a special case, the so called bilevel convex programming problem [8, 14, 17] where some variables are restrict to be in the solution-set of a parametric convex optimization problem. Material on the MPEC problem and its applications can be found in [1, 6]. Problem (P) is called a linear mathematical programming problem with equilibrium constraints LMPEC if D, $C(x)$ are polyhedral convex sets for every x, and f, F are affine functions.

The MPEC problem, even in linear case, is known to be a difficult multi-extremal problem because of its nested structure. Few numerical methods have been proposed for solving the MPEC problem (P) (see e.g. [2-7, 11-13]). All of the existing methods can compute only a local solution or a stationary point. To our knowledges, to date no method has been developed for solving the MPEC or LMPEC problem globally.

This work is an attempt in developing solution methods for globally solving LMPEC problem. By using Kuhn-Tucker theorem for the variational inequality constraints, we reformulate the LMPEC problem as an ordinary linear program with an additional complementarity constraint. For globally solving the latter problem we propose two branch-and-bound algorithms. The first algorithm uses a simplicial subdivision accompanied with a decoupling technique for bounding. The second one uses a binary tree defined according to the sign (zero or positive) of the dual variables appearing in the complementarity condition. The branching in both algorithms takes place in the space of the dual variables whose dimension is just equal to the number of the constraints of the inner variational inequality. Preliminary computational experiences and results show that the algorithm using binary tree is more efficient than the simplicial subdivision algorithm. A lot of randomly generated problems up to twenty five - dual variables are solved by the binary tree algorithm on a PC Pentium II computer.

The remaining of the paper is organized as follows. The next section contains preliminaries including the dual formulation of the problem LMPEC. Section 3 describes the algorithm using simplicial subdivision. Section 4 is devoted to description of the binary tree-algorithm. We close the paper with some preliminary computational experiences and results.

2. Preliminaries

Throughout the paper we suppose that

$$C(x) \equiv \{y \in R^m : g(x,y) := Ax + By + b \geq 0\}, \tag{2}$$

$$F(x,y) \equiv Px + Qy + q \tag{3}$$

where $b \in R^l$, $q \in R^m$ and A, B, P, Q are appropriate given matrices. By applying the Kuhn-Tucker theorem for the linear variational inequality (VI(x)) we can see that LMPEC problem is equivalent to the problem

$$f_* := \min f(x,y) \tag{CP}$$

subject to

$$(x,y) \in D, \tag{4}$$

$$Px + Qy + q - B^T\lambda = 0, \tag{5}$$

$$Ax + By + b \geq 0, \tag{6}$$

$$\lambda \geq 0,\ \lambda^T(Ax + By + b) = 0 \tag{7}$$

in the sense that if the pair (x,y) is a global minimizer of (P) then for any λ satisfying (5) and (7) the triple (λ, x, y) is a global minimizer of (CP); conversely, if the triple (λ, x, y) is a global minimizer of (CP), then the pair (x,y) is a global minimizer of (P). As usual we shall call λ dual variables and (x,y) primal variables.

Solving LMPEC thus amounts to solving Problem (CP). The main difficulty in the latter problem is the complementarity constraint (7). We note that when $\lambda = 0$, Problem (CP) becomes

$$f_0 := \min\ f(x,y) \tag{CP0}$$

subject to

$$(x,y) \in D$$
$$Px + Qy + q = 0,\ Ax + By + b \geq 0.$$

When f is linear and D is a polyhedral convex set given by a system of linear inequalities and/or equalities, Problem (CP0) is an ordinary linear program that can be solved by techniques of linear mathematical programming. Thus we focus on the difficult case when $\lambda \neq 0$. In this case Problem (CP) takes the form

$$f_1 := \min f(x,y) \tag{CP1}$$

subject to

$$(x,y) \in D,$$
$$Px + Qy + q - B^T\lambda = 0,$$
$$\lambda \geq 0,\ \lambda \neq 0,\ Ax + By + b \geq 0,$$
$$\lambda^T(Ax + By + b) = 0.$$

Clearly, $f_* = \min\{f_0, f_1\}$ (as usual we take f_0 or $f_1 = +\infty$ if (CP) or (CP1) is infeasible).

3. A Simplicial subdivision branch-and-bound algorithm

The branch-and bound technique are widely used in integer programming as well as in global optimization. Branch-and-bound algorithms differ from each other by rules they use for branching and bounding. In order to solve Problem (CP1) we propose two branch-and-bound algorithms. The first algorithm uses an adaptive simplicial subdivision accompanied with a decoupling technique for bounding. The second algorithm uses a binary tree defined according to the sign (zero or positive) of the dual variables. The branching in both algorithms takes place in the λ-space whose dimension, in general, is much less than those of the x and y-spaces.

To be precise, let e^i ($i = 1, ..., \ell$) be the ith unit vector of R^ℓ, and S_1 be the convex hull of e^i $(i = 1, ..., \ell)$. Thus S_1 is the $\ell - 1$-simplex whose vertices are $e^1, ..., e^\ell$. Let S be a fully dimensional subsimplex of S_1, and C_S be the polyhedral cone vertexed at the origin whose extreme edges are halflines passing the vertices of S. Clearly C_S is a subcone of the cone R^ℓ_+ (the nonnegative orthant). Consider Problem $(CP1)$ restricted to this cone. That is

$$\text{(CPS)} \qquad f(S) := \min f(x, y)$$

subject to

$$(x, y) \in D,\ Ax + By + b \geq 0,$$
$$Px + Qy + q - B^T\lambda = 0,$$
$$\lambda^T(Ax + By + b) = 0, \lambda \in C_S, \lambda \neq 0.$$

Clearly if $S = S_1$ Problems (CPS) and (CP1) coincide.

Note that if $v > 0$ for every $v \in V(S)$, then $\lambda > 0$ for every $\lambda \in C_S \setminus \{0\}$. In this case Problem (CPS) is reduced to the following two linear programs, one corresponds to $\lambda \neq 0$, the other to $\lambda = 0$:

$$f(S) := \min f(x, y)$$

subject to

$$(x, y) \in D,\ Ax + By + b = 0.$$
$$Px + Qy + q - B^T\lambda = 0,\ \lambda \in C_S, \lambda \neq 0,$$

and

$$f(S) := \min f(x, y)$$

subject to

$$(x, y) \in D,\ Ax + By + b \geq 0,$$
$$Px + Qy + q = 0.$$

In the general case, corresponding to (CPS) we consider the relaxed problem

$$\text{(RCPS)} \qquad \beta(S) := \min f(x, y)$$

subject to

$$(x, y) \in D, Ax + By + b \geq 0,$$
$$Px + Qy + q - B^T z = 0, z \in C_S,$$
$$\lambda^T(Ax + By + b) = 0, \lambda \in S.$$

As usual we set $\beta(S) = +\infty$ if this problem has nofeasible solution. In this case the simplex S is not of interest.

The decoupling by introducing the variable z in the relaxed problem (RCPS) has the following advances.

• The variable λ now can be kept in the unit simplex S_1.

• Solution of Problem (RCPS) gives conditions indicating when the lower bound $\beta(S)$ is exact. In an other word when the partition set S can be deleted from further consideration. Otherwise it suggests an adaptive simplicial subdivision.

• Problem (RCPS) can be solved by using only linear programming methods. Namely we have the following lemma which is the core of the algorithm to be described below.

Lemma 1. (i) *If* (λ, x, y) *is feasible for Problem* (CPS), *then* $(\frac{\lambda}{\sum_{i=1}^{\ell} \lambda_i},\ x, y, \lambda)$ *is feasible for* $(RCPS)$ *and* $\beta(S) \leq f(S)$.

(ii) *If* $(\lambda^S, x^S, y^S, z^S)$ *is optimal for (RCPS) and the condition*

$$(*) \qquad (z^S)^T(Ax^S + By^S + b) = 0$$

is satisfied, then (z^S, x^S, y^S) *is optimal for (CPS) and* $\beta(S) = f(S)$.

(iii) *If Problem (RCPS) is solvable, then there is an optimal solution* (λ, x, y, z) *such that* $\lambda \in V(S)$ *(the set of the vertices of* S*).*

Proof. (i) is obvious from the definition of Problems (CPS) and (RCPS).

(ii) Suppose that z^S satisfies the complementarity condition

$$(z^S)^T(Ax^S + By^S + b) = 0.$$

Then (z^S, x^S, y^S) is feasible for (CPS). Hence (z^S, x^S, y^S) is optimal for (CPS) and therefore $\beta(S) = f(S)$.

(iii) Let $(\lambda^S, x^S, y^S, z^S)$ be an optimal solution to (RCPS). Let $v^1, ..., v^\ell$ denote the vertices of the simplex S. If $\lambda^S \notin V(S)$, then there exist

$$0 \leq \alpha_i \leq 1, (i = 1, ..., \ell),\ \sum_{i=1}^{\ell} \alpha_i = 1$$

such that

$$\lambda^S = \sum_{i=1}^{\ell} \alpha_i v^i.$$

Since $(\lambda^S)^T(Ax^S + By^S + b) = 0$, replacing λ^S by

$$\lambda^S = \sum_{i=1}^{\ell} \alpha_i v^i$$

we obtain

$$\sum_{i=1}^{\ell} \alpha_i (v^i)^T (Ax^S + By^S + b) = 0. \tag{8}$$

Since

$$v^i \geq 0,\ \ \alpha_i \geq 0\ \forall i$$

and

$$Ax^S + By^S + b \geq 0,$$

it follows that

$$(v^i)^T(Ax^S + By^S + b) = 0 \ \text{ for every } \ \ i. \tag{9}$$

Thus (v^i, x^S, y^S, z^S) is feasible (RCPS), and therefore it is also optimal for (RCPS). □

Remark 1. Clearly, if $z^S = \alpha\lambda^S$ for some $\alpha \geq 0$, then from $(\lambda^S)^T(Ax^S + By^S + b) = 0$ follows $(z^S)^T(Ax^S + By^S + b) = 0$.

Remark 2. Suppose $S_1 = \cup_{i\in I} S_i$. Let S_k $(k \in I)$ such that

$$\beta(S_k) = \min\{\beta(S_i) :\ i \in I\}$$

where $\beta(S_i)$ is the optimal value of the relaxed problem $(RCPS_i)$. Let α_k be an upper bound for the optimal value of Problem (CP1). If $\beta(S_k) = \alpha_k$, then α_k is the global optimal value of (CP1).

Remark 3. Let S be a subsimplex of S_1. Consider the problem

$$\delta(S) := \min \lambda^T(Ax + By + b) \tag{10}$$

subject to

$$(x, y) \in D,\ \ Ax + By + b \geq 0,$$
$$Dx + Qy + q - B^T z = 0, z \in C_S, \lambda \in S.$$

Clearly that if $\delta(S) > 0$, then the relaxed problem (RCPS) has nonfeasible point. The subsimplex S then can be eliminated from further consideration. Note that the objective function (10) is a bilinear function which implies that this problem attains its optimal value at a point (λ, x, y, z) such that $\lambda \in V(S)$. Thus $\delta(S)$ can be computed by solving linear programs, one for each vertex of S.

Computing Upper Bound

The feasible domain of Problem (CPS), in general, is nonconvex. However unlike general mathematical programming problems having nonconvex feasible domains, a feasible point of this problem can be computed with a reasonable effort due to the specific structure of the complementarity constraint $\lambda^T(Ax+By+b) =$

0. Without any additional assumption on the matrices A and B, a feasible point, thereby an upper bound, can be computed, for example, as follows.

Let $(\lambda^S, x^S, y^S, z^S)$ be an optimal solution to the relaxed problem (RCPS). Let

$$J_+(\lambda^S) := \{j : \lambda_j^S > 0\}.$$

Solve the linear program

(UCPS) $$\alpha(S) := \min f(x, y)$$

subject to

$$(x, y) \in D,\ (Ax + By + b)_j \geq 0\ j \notin J_+(\lambda^S)$$
$$Px + Qy + q - B^T\lambda^S = 0,$$
$$(Ax + By + b)_j = 0, (j \in J_+(\lambda^S)).$$

Clearly, if (x, y) is an optimal solution of this linear program, then (λ^S, x, y) is feasible for (CP1) , and therefore $\alpha(S)$ is an upper bound for f_1.

A Simplicial Subdivision

A subdivision is said to be simplicial subdivision if its every partition set is simplex. Simplicial subdivisions are often used in global optimization (see e.g. [4]). The main advantage of a simplicial subdivision is that the number of the vertices of every generated partition set is not larger than the dimension of the space where the subdivision takes place. In the algorithm 1 to be described below we shall use the following simplicial subdivision [4].

Suppose that S is the simplex to be subdivided. Let $(v^1, ..., v^\ell)$ be the vertices of S. We may assume that Problem (RCPS) is solvable because if it is infeasible, then S can be deleted. Let $(\lambda^S, x^S, y^S, z^S)$ be an optimal solution of Problem (RCPS).

Note that if the condition (*) in Lemma 1 is fulfilled, then (z^S, x^S, y^S) is optimal for (CPS) and therefore $\beta(S) = \alpha(S)$. In this case the simplex S can also be deleted. So we may assume that this condition does not hold. Let $\pi(z^S)$ be the point where the halfline connecting the origin and z^S meets S. Since the condition (*) does not hold, $\lambda^S \neq \pi(z^S)$ (Remark 1). Let $\omega^S := (\pi(z^S) + \lambda^S)/2$. Since $\omega^S \in S$, we have

$$\omega^S = \sum_{i=1}^{\ell} t_i v^i,\ \sum_{i=1}^{\ell} t_1 = 1,\ t_i \geq 0,\ (i = 1, ..., \ell).$$

Let $I(\omega^S) := \{i : t_i > 0\}$. We then subdivide S into subsimplices S_i, where S_i $(i \in I(\omega^S))$ is obtained from S by replacing the vertex v^i of S with ω^S. In the literature this subdivision is often called *radial subdivision*. The points $\pi(z^S)$ and λ^S are called subdivision points for S (see e.g.[4]). Note that if $\pi(z^S)$ and λ^S are two adjacent vertices of S and the edge connecting $\pi(z^S)$ and λ^S is longest, then this is an exhaustive bisection process. We recall that a subdivision process is said to be exhaustive if its every infinite sequence of nested partition sets shinks to a singleton [4, 16].

Definition 1. The above defined radial simplicial subdivision is said to be *locally exhaustive* if any two infinite sequences of the subdivision points for an infinite nested partition sets have a common cluster point.

Clearly, every exhaustive simplicial subdivision is locally exhaustive.

ALGORITHM 1. Choose a tolerance $\varepsilon \geq 0$.

Computing lower bound $\beta(S_1)$ by solving the linear program $(RCPS_1)$. If this problem has nonfeasible point, then Problem (CP1) has nonfeasible point too. The algorithm terminates. Otherwise, let $(\lambda^1, x^1, y^1, z^1)$ be the obtained solution.

If

$$\langle z^1, Ax^1 + By^1 + b\rangle = 0,$$

then let $\alpha_1 = \beta_1 = \beta(S_1)$ and $(\xi^1, u^1, v^1) := (z^1, x^1, y^1)$.

Otherwise, let $\alpha_1 = f(x^1, y^1)$ where (λ^1, x^1, y^1) is a feasible point of (CP1) known in advance (if no feasible point is known take $\alpha_1 = +\infty$).

Take

$$\Gamma_1 := \begin{cases} \{S_1\} & \text{if } \alpha_1 - \beta_1 > \varepsilon(|\alpha_1| + 1), \\ \emptyset & \text{otherwise.} \end{cases}$$

Let $k := 1$.

Iteration k ($k = 1, 2...$) At each iteration k we have a family Γ_k of simplices. To each $S \in \Gamma_k$ we associate a real number $\beta(S)$ which serves as a lower bound for the optimal value $\alpha(S)$, and we have an optimal solution $(\lambda^S, x^S, y^S, z^S)$ of (RCPS). Also we may have an feasible point (ξ^k, u^k, v^k) (incumbent) and an upper bound $\alpha_k = f(u^k, v^k)$.

a) If $\Gamma_k = \emptyset$, then terminate: (ξ^k, u^k, v^k) is an ε-global optimal solution to (CP1).

b) If $\Gamma_k \neq \emptyset$, then choose S_k such that

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

Subdivide S_k into subsimplices $S_{ki}, i \in I(\omega^{S_k})$ according to the simplicial subdivision described above.

Compute $\beta(S_{ki})$ by solving linear programs $(RCPS_{ki})$ $(i \in I(\omega^{S_k}))$.

Compute upper bounds $\alpha(S_{ki})$ by solving the linear programs $(UCPS_{ki})$ $(i \in I(\omega^{S_k}))$. Use $\alpha(S_{ki})$ to update currently best upper bound α_{k+1} and the incumbent $(\xi^{k+1}, u^{k+1}, v^{k+1})$. Set

$$\Gamma'_{k+1} := (\Gamma_k \setminus \{S_k\}) \cup \{S_{ki} : i \in I(\omega^{S_k})\}.$$

$$\Gamma_{k+1} := \{S \in \Gamma'_{k+1} : \alpha_{k+1} - \beta(S) > \varepsilon(|\alpha_{k+1}| + 1)\}.$$

Increase k by 1 and go to iteration k.

Remark 4. In order to enhance convergence, in Algorithm 1, for each generated simplex S we may compute $\delta(S)$ by solving the problem (10) (see Remark 3). If $\delta(S) > 0$, then the simplex S can be deleted from further consideration.

Theorem 1. (i) *If the algorithm terminates at some iteration k, then α_k is an ε-global optimal value and (ξ^k, u^k, v^k) is an ε-global optimal solution to Problem (CP1).*

(ii) *Suppose that the sequence $\{(z^k, x^k, y^k)\}$ is bounded and the subdivision used in the algorithm is locally exhaustive. Then if the algorithm does not terminate we have $\beta_k \nearrow f_1$ and the sequence $\{(z^k, x^k, y^k)\}$ has a cluster point that solves Problem (CP1). Moreover if the subdivision is exhaustive, then any cluster point of (z^k, x^k, y^k) solves (CP1).*

Proof. (i) If the algorithm terminates at iteration k, then $\Gamma_k = \emptyset$. This means that $\alpha_k - \beta_k \leq \varepsilon(|\alpha_k| + 1)$. Since α_k is an upper bound and β_k is a lower bound for the optimal value of (CP1), it follows that α_k is an ε-global optimal value. Thus the currently best feasible point (ξ^k, u^k, v^k) is an ε-global optimal solution to (CP1).

(ii) If the algorithm never terminates, then it generates a nested sequence of simplices, say $\{S_q\}$. Let $(\lambda^q, x^q, y^q, z^q)$ be the obtained solution of $(RCPS_q)$. Then

$$\beta_q = \beta(S_q) = f(x^q, y^q) \leq f_1.$$

Since the sequences $\{x^q\}$ and $\{y^q\}$ are bounded, by taking subsequences if necessary, we may assume that $x^q \to x^*$, and $y^q \to y^*$ as $q \to \infty$.

Note that $\{\beta_q\}$ is an increasing sequence, letting $q \to \infty$ we get

$$\beta_* = \lim \beta_q \leq f(x^*, y^*) \leq f_1. \tag{11}$$

Since the subdivision, by the assumption, is locally exhaustive, the sequences $\{\pi(z^q)\}$ and $\{\lambda^q\}$ has a common cluster point, say λ^*. For simplicity of notation, by taking subsequences if necessary, we may assume that $\lim_q \lambda^q = \lim_q \pi(z^q) = \lambda^*$.

Since $(\lambda^q, x^q, y^q, z^q)$ is optimal for the relaxed problem (RCPS), we have

$$(x^q, y^q) \in D, Ax^q + By^q + b \geq 0,$$

$$Px^q + Qy^q + q - B^T z^q = 0, z^q \geq 0,$$

$$(\lambda^q)^T (Ax^q + By^q + b) = 0, \lambda^q \geq 0.$$

Letting $q \to \infty$ we obtain in the limit that

$$(x^*, y^*) \in D, Ax^* + B^* y + b \geq 0,$$

$$Px^* + Qy^* + q - B^T z^* = 0, z^* \geq 0,$$

$$(\lambda^*)^T (Ax^* + By^* + b) = 0, \lambda^* \geq 0.$$

This and $z^* = \alpha_* \lambda^*$ for some $\alpha_* \geq 0$ imply that (z^*, x^*, y^*) is feasible for (CP1), and therefore $f(x^*, y^*) \geq f_1$. Combining this with (11) we have

$$\beta_* = f(x^*, y^*) = f_1$$

which means that β_* is the optimal value and (z^*, x^*, y^*) is a global optimal solution to (CP1).

Now assume that the subdivision is exhaustive. Let (z^*, x^*, y^*) be any cluster point of the sequence $\{(z^k, x^k, y^k)\}$, and let $\{(z^q, x^q, y^q)\}$ be the subsequence that converges to (z^*, x^*, y^*). Since S_q is the selected partition set at iteration q, it is easy to see (see also [9, 10]) that one can choose a subsequence of $\{S_q\}$, which, for simplicity of notation, we also denote by $\{S_q\}$, and a nested subsequence $\{S_j\}$ of $\{S_k\}$ such that for every q there is j_q satisfying $S_q \subset S_{j_q}$. Since the sequence $\{S_j\}$ is nested, by exhaustiveness we have $S_j \to \bar{\lambda}$. This implies $S_q \to \bar{\lambda}$. Hence $\lambda^* = \bar{\lambda}$. By the same way as above we can show that (λ^*, x^*, y^*) is feasible for (CP1), and therefore $f(x^*, y^*) = f_1$. □

Remark 5. The sequence $\{(z^k, x^k, y^k)\}$ is bounded if the set

$$\{(\lambda, x, y) : (x, y) \in D, \lambda \geq 0, Ax + By + b \geq 0, Px + Qy + q - B^T\lambda = 0\}$$

is bounded.

Remark 6. Theoretically, in the above algorithm it may happens that no one of feasible points are found (Problems $(UCPS_k)$ are infeasible). Then $\alpha_k = +\infty$ for every k. In this case, we terminate the algorithm if either

$$(z^k)^T(Ax^k + By^k + b) \leq \varepsilon_0,$$

or

$$\|Px^k + Qy^k + q - B^T\lambda^k\| \leq \varepsilon_1$$

where $\varepsilon_0 > 0$ and $\varepsilon_1 > 0$ are given tolerance. In the first case, since $(\lambda^k, x^k, y^k, z^k)$ is an optimal solution of Problem $(RCPS_k)$, we have $f(x^k, y^k) \leq f_1$ and

$$(x^k, y^k) \in D, Ax^k + By^k + b \geq 0,$$
$$Px^k + Qy^k + q - B^T z^k = 0, z^k \geq 0,$$
$$(z^k)^T(Ax^k + By^q + b) \leq \varepsilon_0,$$

i.e., all constraints are satisfied except the complementarity constraint is satisfied with ε_0-tolerance. Thus the trip (z^k, x^k, y^k) can be regarded as an approximate solution.

Similarly, if

$$\|Px^k + Qy^k + q - B^T\lambda^k\| \leq \varepsilon_1,$$

then the trip (λ^k, x^k, y^k) satisfies all constraints except the constraint

$$Px + Qy + q - B^T\lambda = 0$$

is satisfied with an ε_1-tolerance. In this case the trip (λ^k, x^k, y^k) can also be regarded as an approximate solution.

Remark 7. The weak exhaustiveness assumption in the convergence theorem is fulfilled if the simplicial subdivision is defined according to the following rule.

Rule 1. Let $N \geq 1$ be a given natural number. Let S_k be the simplex to be subdivided at iteration k. If k is a multiplier of N, then we use the radial subdivision with ω^k being the midpoint of a longest edge of S_k. Otherwise ω^k is the midpoint of the segment connecting λ^{S_k} and $\pi(z^{S_k})$.

4. Branching and bounding by binary tree

It is well known that the binary tree has been successfully applied for solving a lot number of zero-one mathematical programming problems [15]. In this section we shall use the complementarity condition to construct a binary tree which allow us to derive a branch-and-bound algorithm for solving Problem (CP). Here we do not distinguish two cases $\lambda \neq 0$ and $\lambda = 0$ as the previous section. By f_* we will denote the optimal value of Problem (CP).

First let us define a binary tree as follows:

The tree is defined according to the sign of the dual variables $\lambda_1, ..., \lambda_\ell$. To each node of the tree we associate a dual variable by fixing it to be zero or positive. Every node has exactly two branches. The node corresponding to the variable λ_j has two branches: one corresponds to $\lambda_{j+1} = 0$, the other to $\lambda_{j+1} > 0$. The root has two branches corresponding to $\lambda_1 = 0$ and $\lambda_1 > 0$. Note that a variable λ_j may correspond to one or more nodes, but a node corresponds to exactly one variable. We agree to call a node *partition set*. The initial partition set T_1 is the root of the tree. Since the number of variables λ_j is finite, the binary tree is finite too, i.e., it has only a finite number of nodes.

Let $P(T)$ denote the path from the root T_1 to the node T, and let $J(T) \subset \{1, ..., \ell\}$ denode the set of indeces that correspond to the nodes belonging to the path $P(T)$. Let

$$J_0(T) := \{j \in J(T) : \lambda_j = 0\}$$
$$J_1(T) := \{j \in J(T) : \lambda_j > 0\}.$$

Since at the root the sign of every variable λ_j is free, $J_0(T_1) = J_1(T_1) = \emptyset$.

Let (CPT) denote the problem (CP) restricted to the partition set T, i.e.,

$$\text{(CPT)} \qquad f(T) := \min f(x, y)$$

subject to

$$(x, y) \in D,$$
$$Px + Qy + q - B^T\lambda = 0, \ Ax + By + b \geq 0,$$
$$\lambda^T(Ax + By + b) = 0, \lambda \geq 0, \ \lambda_j = 0 \ \ j \in J_0(T),$$
$$(Ax + By + b)_j = 0, \ j \in J_1(T).$$

Since $J_0(T_1) = J_1(T_1) = \emptyset$, Problem (CPT_1) is just (CP).

To obtain a lower bound for $f(T)$ we consider the relaxed problem

$$\text{(RCPT)} \qquad \beta(T) := \min f(x, y)$$

subject to

$$(x, y) \in D,$$
$$Px + Qy + q - B^T\lambda = 0, \ (Ax + By + b)_j \geq 0 \ \ j \notin J_1(T)$$
$$\lambda \geq 0, \lambda_j = 0 \ \ j \in J_0(T), (Ax + By + b)_j = 0 \ \forall j \ \ \in J_1(T).$$

Since the feasible domain of (CPT) is contained in that of (RCPT), we have $\beta(T) \leq f(T)$. In particular $\beta(T_1) \leq f(T_1) = f_*$. Note that $\beta(T)$ is an upper bound for f_* if T is an end node of the free. To obtain an upper bound for the optimal value of Problem (CP) we solve the following problems

(UCPT) $$\alpha(T) := \min f(x,y)$$

subject to

$$(x,y) \in D,$$
$$Px + Qy + q - B^T\lambda = 0,\ (Ax + By + b)_j \geq 0 \ \ j \in J_0(T)$$
$$(Ax + By + b)_j = 0, j \notin J_0(T),\ \lambda \geq 0, \lambda_j = 0\ j \in J_0(T),$$

or

(UCPT) $$\alpha(T) := \min f(x,y)$$

subject to

$$(x,y) \in D,$$
$$Px + Qy + q - B^T\lambda = 0,\ (Ax + By + b)_j \geq 0 \ \ j \notin J_1(T)$$
$$(Ax + By + b)_j = 0, j \in J_1(T), \lambda \geq 0, \lambda_j = 0\ j \ \notin J_1(T).$$

Clearly, any solution of one of these problems is feasible for Problem (CP). A node (partition set) T is deleted (dead) if $\beta(T) \geq \alpha$ where α is an upper bound for the optimal value of (CP). If a node corresponding to some variable, say λ_j, is not deleted, it is branched (bisected) into two nodes by setting $\lambda_{j+1} = 0$ and $\lambda_{j+1} > 0$.

Having this binary tree the algorithm can be described as follows.

ALGORITHM 2. Let the tolerance $\varepsilon \geq 0$ be chosen in advance.

Compute the lower bound $\beta(T_1)$ by solving linear program $(RCPT_1)$. Let (λ^1, x^1, y^1) be the obtained solution. If

$$\langle \lambda^1, Ax^1 + By^1 + b\rangle = 0,$$

then let $\alpha_1 = \beta_1 = \beta(T_1)$ and $(\xi^1, u^1, v^1) := (\lambda^1, x^1, y^1)$.

Otherwise, let $\alpha_1 = f(x^1, y^1)$ where (λ^1, x^1, y^1) is a feasible point of (CP_1) known in advance. $\alpha_1 = +\infty$).

Take

$$\Gamma_1 = \begin{cases} \{T_1\} & \text{if} \quad \alpha_1 - \beta_1 > \varepsilon(|\alpha_1| + 1) \\ \emptyset & \text{otherwise.} \end{cases}$$

Let $k := 1$

Iteration k ($k = 1, 2...$)

a) If $\Gamma_k = \emptyset$, then terminate: (ξ^k, u^k, v^k) is an ε-global optimal solution to (CP).

b) If $\Gamma_k \neq \emptyset$, then choose T_k such that

$$\beta_k := \beta(T_k) = \min\{\beta(T) : T \in \Gamma_k\}.$$

Branch T_k into two nodes T_{k1} and T_{k2} by setting $\lambda_{i+1} = 0$ for the node T_{k1} and $\lambda_{i+1} > 0$ for T_{k2}, where λ_i is the variable corresponding to the node T_k.

Compute $\beta(T_{ki})$ by solving linear programs $(RCPT_{ki})$ $(i = 1, 2)$.

Compute upper bounds $\alpha(T_{k1})$ and $\alpha(T_{k2})$ by solving linear programs $(UCPT_{k1})$ and $(UCPT_{k2})$. Use $\alpha(T_{k1})$ and $\alpha(T_{k2})$ to update the incumbent $(\xi^{k+1}, u^{k+1}, v^{k+1})$ and the currently best upper bound $\alpha_{k+1} = f(u^{k+1}, v^{k+1})$.

Set

$$\Gamma'_{k+1} := (\Gamma_k \setminus T_k) \cup \{T_{k1}, T_{k2}\},$$
$$\Gamma_{k+1} := \{T \in \Gamma'_{k+1} : \alpha_{k+1} - \beta_{k+1} > \varepsilon(|\alpha_{k+1}| + 1)\}.$$

Increase k by 1 and go to iteration k.

Since the binary tree has a finite number of nodes, unlike Algorithm 1, Algorithm 2 always terminates after a finite iterations yielding an ε- global optimal solution to Problem (CP).

5. Computational experiences and results

In order to obtain a preliminary evaluation of the performance of the proposed algorithms, we have written computer codes by PASCAL TURBO 7.0 that implements the algorithms. The codes used the ordinary simplex method for solving the linear programs called for by the algorithms. We use the codes to solve hundred randomly generated problems on a Pentium II personal computer. For all test problems we take $\varepsilon = 10^{-4}$. The computed results are reported in Table 1 for Algorithm 1 and in Table 2 for Algorithm 2.

In the tables we use the following headings:

- ℓ: the number of the dual variables which equals to the number of constraints (without nonnegative one) of the inner variational inequality,

- n, m: the number of variables x and y respectively,

- *iter* : the average number of the iterations

- *time*: the average CPU time (in second),

- σ the average deviation (in time).

Table 1

Prob.	l	n	m=p	iter.	time(s)	σ
1-10	3	30	10	1.0	0.731	0.0727
11-20	4	30	10	46.7	115.389	330.295
21-30	5	30	10	36.1	103.633	190.338
31-36	6	30	10	148.3	283.85	500.628
37-44	3	20	30	37.4	285.166	456.730
45-53	4	20	30	110.7	1626.383	3053.979

The results in the tables show that Algorithm 1 is efficient for problems where ℓ is small (≤ 6). Algorithm 2 is much more efficient than Algorithm 1. In fact, Algorithm 2 can be used to solve problems up to 25-dual variables.

As it is expected, the running time is much more sensitive to growth in ℓ, the number of constraints of the follower variational inequality, than to growth in the numbers of the x-variables (parametric) and the y-variables, the number of the constraints of the leader problem.

Table 2

Prob.	l	n	m=p	iter.	time(s)	σ
1-10	7	50	30	11.4	48.170	31.536
11-20	10	50	30	45.2	217.924	295.999
21-30	15	20	35	123.5	678.846	402.214
31-40	20	20	20	532.3	1963.848	2030.305
41-46	25	25	25	1736.2	12838.056	7877.376

Remark. A very short version without proofs and detail description of the algorithms will appear as a short communication in Vietnam Journal of Mathematics.

References

[1] F. Facchinei, H. Jiang and L. Qi, *A smoothing method for mathematical programs with equilibrium constraints*, Math. Program. **85** (1999), 107-134.

[2] M. Fukushima, Z-Q Luo, and J. S. Pang, *A globally convergent sequential quadratic programming algorithm for mathematical programs with linear complementarity constraints*, Comput. Optim. Appl. (to appear).

[3] M. Fukushima and J. S. Pang, *Some feasibility issues in mathematical programs with equilibrium constraints*, SIAM J. Optim. **8** (1998), 673-681.

[4] R. Horst and H. Tuy, *Global Optimization (Deterministic Approach)* 3rd Edition, Springer Verlag Berlin, Germany, 1996.

[5] H. Jiang and D. Ralph, *OPECgen, a MATLAB generator for mathematical programs with quadratic objectives and affine variational inequality constraints*, Comput. Optim. Appl. **13** (1999), 25-59.

[6] Z. Q. Luo, J. S. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, New York, 1997.

[7] Z. Q. Luo, J. S. Pang, and D. Ralph, *Piecewise Sequential Quadratic Programming for Mathematical Program with nonlinear complementarity Constraints*, in: Multilevel Optimization, Algorithm, Complexity and Application, A, Migdalas et al. (Eds), *Kluwer Academic Publisher*, (1998), 209-229.

[8] A. Migdalas, P. M. Padalos and P. Varbrand , *Multilevel Optimization: Algorithms and Applications*, Kluwer Academic Publishers, 1998.

[9] Le D. Muu and W. Oettli, *A* method for minimizing a convex-concave function over a convex set, J. Optimization Theory Appl. **70** (1991), 377-384.

[10] Le D. Muu, *An algorithm for solving convex programs with an additional convex-concave constraint*, Math. Program. **61** (1993), 75-87.

[11] J. V. Outrata, *On optimization problems with variational inequality constraints*, SIAM J. Optim. **4** (1994), 340-357.

[12] J. V. Outrata and J. Zowe, *A numerical approach to optimization problems with variational inequality constraints*, Math. Program. **68** (1995), 105-130.

[13] J. S. Pang, *Complementarity Problems, in HandBook of Global Optimization*, R. Horst and P. Pandalos (Eds), Kluwer Academic Publishers: Boston, (1995), 271-338.

[14] N. V. Quy and Le. D. Muu, *On penalty function method for dual form of a class of non convex constrained optimization problems, Application to Linear Bilevel Programming*. Vietnam J. Math. (to appear).

[15] M. M. Syslo, V. Deo, and J. S. Kowallk, *Discrete optimization algorithms with Pascal programs*, Prentice-Hall, New Jersey 1983.

[16] N. V. Thoai and H. Tuy, *Convergent algorithms for minimizing a concave function*, Math. Oper. Res. **5** (1980), 556-566.

[17] L. Vicente, G. Savard, and J. Judice, *Descent approaches for quadratic bilevel programming*, J. Optimization Theory Appl. **81** (1994), 379-399.

Hanoi Institute of Mathematics
P.O. Box 631, Bo Ho 10000, Hanoi, Vietnam

E-mail address: ldmuu@thevinh.ncst.ac.vn