J Glob Optim (2007) 38:527–554
DOI 10.1007/s10898-006-9093-1

A novel approach to Bilevel nonlinear programming

H. Tuy · A. Migdalas · N. T. Hoai-Phuong

Received: 6 September 2006 / Accepted: 6 September 2006 / Published online: 16 January 2007

Abstract Recently developed methods of monotonic optimization have been applied successfully for studying a wide class of nonconvex optimization problems, that includes, among others, generalized polynomial programming, generalized multiplicative and fractional programming, discrete programming, optimization over the efficient set, complementarity problems. In the present paper the monotonic approach is extended to the General Bilevel Programming GBP Problem. It is shown that (GBP) can be transformed into a monotonic optimization problem which can then be solved by "polyblock" approximation or, more efficiently, by a branch-reduce-and-bound method using monotonicity cuts. The method is particularly suitable for Bilevel Convex Programming and Bilevel Linear Programming.

Keywords Bilevel nonlinear programming · Bilevel convex programming · Bilevel linear programming · Leader and follower game · Monotonic optimization · Polyblock approximation · Branch-reduce-and-bound method · Monotonicity cuts

AMS subject classification 90C26 · 65K05 · 90C20 · 90C30 · 90C56 · 78M50

1 Introduction

In hierarchical systems involving two levels of decision making with different, sometimes conflicting, objectives, the following optimization problem may arise. The higher

H. Tuy (✉) · N. T. Hoai-Phuong
Institute of Mathematics, Hanoi, Vietnam
e-mail: htuy@math.ac.vn

A. Migdalas
Technical University of Crete, Chania, Crete, Greece
e-mail: migdalas@ergasya.tuc.gr

N. T. Hoai-Phuong
e-mail: htphuong@math.ac.vn

level (leader) controls a first set of decision variables (for instance transfer prices, resources allocation, capital investment channelling, etc...), while the lower level (follower) controls a second set of decision variables (activity level, production volume, technology alternatives, etc...). To each decision made by the higher level, the lower level responds by a decision optimizing its objective function over a constraint set which depends upon the decision of the higher level. Assuming that the higher level perfectly knows the reaction of the lower level to each of its decisions, the problem is to find an optimal decision of the higher level, i.e. a decision that ensures the best value of the overall objective function under these conditions. When the objective functions of the leader and the follower as well as the constraints of the lower level are linear, this problem is called a *bilevel linear program* (BLP).

Originally formulated and studied as a mathematical program by Bracken and McGill (8,9), bilevel and, more generally, multilevel optimization has become a subject of extensive research during the last two decades due to numerous applications in diverse fields: economic development policy, agriculture economics, road network design, oil industry regulation, international water systems and flood control, energy policy, traffic assignment, structural analysis in mechanics, etc. Multilevel optimization can also be useful for the study of hierarchical designs in many complex systems, in particular, biological systems (see, e.g. (5,40)).

A special case of (BLP) where the follower's objective function is the negative of the leader's, reduces to the linear max–min problem which is actually equivalent to a linearly constrained convex maximization problem and hence can be solved by methods extensively studied during the last three decades. The general bilevel linear programming problem is, however, more complicated and fraught with pitfalls. Actually, several methods proposed in the literature for its solution turned out to be nonconvergent, or incorrect or convergent to a local optimum (see, e.g. (6)), where some of these errors have been reported). A review of bilevel optimization methods has been given in (34) (see also [38]).

Most solution approaches proposed to date for (BLP) use, directly or indirectly, a reformulation of it as a single level mathematical program which, in most cases, is a linear program with an additional nonconvex constraint. It is this additional nonconvex constraint, of a rather peculiar form, which constitutes the major source of difficulty in this approach and is actually the cause of most of the above mentioned errors.

While the bilevel linear problem has been extensively studied, the literature on general (nonlinear) bilevel problem (GBP) is rather poor. So far most research in this field is limited to convex or quadratic bilevel programming problems, and/or is mainly concerned with finding stationary points and local minima rather than global optimal solutions [2, 3, 7, 10, 23, 26, 31, 39, 41]. For general nonlinear bilevel programs, most investigations have been concentrated on theoretical aspects (11,13,14,21,25). Very few exact methods exist, though some general properties of (GBP), including its relation to multiobjective programming, have been discussed (15,20,21,27).

A specific feature of (GBP) that has been observed in our earlier works (29,30) is that the nonconvex constraint generated by the second level problem is in fact a monotonic constraint. Consequently, (GBP) actually belongs to the realm of monotonic optimization problems as formulated and studied recently in a series of papers (32,33,35,36).

The aim of the present paper is to develop a novel approach to the general bilevel programming problem (GBP), and especially the Bilevel Convex Programming

Problem, based on this monotonic reformulation. First, in Sect. 2, we review some basic concepts and results of monotonic optimization. In Sect. 3 we state (GBP), together with the basic assumptions, and reformulate it as a a single level monotonic optimization problem. Next, in Sect. 4 a solution approach is outlined by specializing a method called polyblock approximation earlier proposed in (32,33) for monotonic optimization. Since the complexity of this algorithm increases rapidly with the dimension of the problem, in Sect. 5 a more suitable method, of the branch and cut type, called branch-reduce-and-bound (BRB) method, is presented. Section 6 is devoted to some implementation issues, while Sect. 7 discusses cases when the algorithm can be modified so as to work in a space of dimension smaller than m and also clarifies the relation of bilevel programming to optimization over the efficient set in multiobjective programming. Finally, the paper closes with some numerical examples illustrating how the method works in practice.

2 Some basic concepts of monotonic optimization

We begin with a review of some basic concepts and results of monotonic optimization as discussed in (32,36).

Let $[a,b] = \{x \in \mathbb{R}^n | \; a \leq x \leq b\}$ be a box in $\mathbb{R}^n$. A function f: $[a,b] \to \mathbb{R}$ is said to be *increasing* if $a \leq z \leq z' \leq b$ implies $f(z) \leq f(z')$; *decreasing* if $a \leq z \leq z' \leq b$ implies $f(z) \geq f(z')$; *monotonic* if it is either increasing or decreasing. Increasing and decreasing functions abound in mathematics, engineering, economics and other fields.

A set $E \subset [a,b]$ is said to be *downward* (or *normal*) if $a \leq x' \leq x \in E$ implies $x' \in E$; *upward* (or *conormal*, or *reverse normal*) if $b \geq x' \geq x \in E$ implies $x' \in E$.

Proposition 1 *If φ: $[a,b] \to \mathbb{R}$ is increasing then the set $G = \{x \in [a,b] | \; \varphi(x) \leq 0\}$ is downward, the set $H = \{x \in [a,b] | \; \varphi(x) \geq 0\}$ is upward.*

Proof Immediate. □

A point z in a downward set $G \subset [a,b]$ is called an *upper boundary point* if $z \in \mathrm{cl}G$ and $z' > z$ implies $z' \notin G$. A point z of an upward set $H \subset [a,b]$ is called a *lower boundary point* if $z \in \mathrm{cl}H$ and $z' < z$ implies $z' \notin H$. The set of upper boundary points of G is called its upper boundary and is denoted by $\partial^+ G$. The set of lower boundary points of H is called its *lower boundary* and is denoted by $\partial^- H$.

If an increasing function $\varphi(x)$ is continuous then the intersection of $G = \{x \in [a,b] | \; \varphi(x) \leq 0\}$ with any line segment joining a point $p \in G$ with a point $q \in [p,b] \setminus G$ is a segment $[p, \pi_p(q)] \subset [p,q]$ such that $\pi_p(q) \in \partial^+ G$. Clearly, $\varphi(\pi_p(q)) = 0$ and

$$\pi_p(q) = q - \bar{\alpha}(q-p) \quad \text{with } \bar{\alpha} = \min\{\alpha | \; \varphi(q - \alpha(q-p)) \leq 0\}, \tag{1}$$

$$= p + \bar{\beta}(q-p) \quad \text{with } \bar{\beta} = \max\{\beta | \; \varphi(p + \beta(q-p)) \leq 0\}. \tag{2}$$

Proposition 2 *The maximum of a continuous increasing function (or the minimum of a continuous decreasing function) over the set $G = \{x \in [a,b] | \; \varphi(x) \leq 0\}$ is achieved on $\partial^+ G$.*

Proof Immediate. □

Given a set $A \subset [a,b]$ the *downward hull* (*normal hull*) of A, written $A^{\rceil}$, is the smallest downward set containing A. The *upward hull* (or *conormal hull*) of A, written $\lfloor A$, is the smallest upward set containing A.

Proposition 3

(1) *The downward hull of a set* $A \subset [a,b] \subset \mathbb{R}^n_+$ *is the set* $A^{\rceil} = \cup_{z\in A}[a,z] = \{y \in [a,b] | \, x \le y$ *for some* $x \in A\}$. *If* A *is compact then so is* $A^{\rceil}$.

(2) *The upward hull of a set* $A \subset [a,b] \subset \mathbb{R}^n_+$ *is the set* $\lfloor A = \cup_{z\in A}[z,b] = \{y \in [a,b] | \, y \le x$ *for some* $x \in A\}$. *If* A *is compact then so is* $\lfloor A$.

Proof It suffices to prove (1) because the proof of (2) is similar. Let $P = \cup_{z\in A}[a,z]$. Clearly P is downward and $P \supset A$, hence $P \supset A^{\rceil}$. Conversely, if $x \in P$ then $x \in [a,z]$ for some $z \in A \subset A^{\rceil}$, hence $x \in A^{\rceil}$ by normality of $A^{\rceil}$, so that $P \subset A^{\rceil}$ and therefore, $P = A^{\rceil}$. If A is compact then A is contained in a ball B centered at 0, and if $x^k \in A^{\rceil}, k = 1,2,\ldots$, then since $x^k \in [a,z^k] \subset B$, there exists a subsequence $\{k_\nu\} \subset \{1,2,\ldots\}$ such that $z^{k_\nu} \to z^0 \in A$, $x^{k_\nu} \to x^0 \in [a,z^0]$, hence $x^0 \in A^{\rceil}$, proving the compactness of $A^{\rceil}$. □

If V is a finite subset of $[a,b]$ then $P = V^{\rceil}$ (the downward hull of V) is called a *polyblock* with *vertex set* V. By Proposition 3, $P = \cup_{z\in V}[a,z]$. A vertex z of a polyblock is called *proper* if there is no vertex $z' \neq z$ "dominating" z, i.e., such that $z' \geq z$. An *improper* vertex or improper element of V is an element of V which is not a proper vertex. Obviously, a polyblock is fully determined by its proper vertex set; more precisely, *a polyblock is the downward hull of its proper vertices*.

Similarly, if V is a finite subset of $[a,b]$ then $Q = \lfloor V$ is called an *upward polyblock* (or *reverse polyblock*) with *vertex set* V. By Proposition 3, $\lfloor V = \cup_{z\in V}[z,b]$. A vertex z of an upward polyblock is called *proper* if there is no vertex $z' \neq z$ "dominated" by z, i.e., such that $z' \le z$. An *improper* vertex or improper element of V is an element of V which is not a proper vertex. Obviously, an upward polyblock is fully determined by its proper vertex set; more precisely, *an upward polyblock is the upward hull of its proper vertices*.

Proposition 4

(1) *The intersection of finitely many polyblocks is a polyblock.*

(2) *The intersection of finitely many upward polyblocks is an upward polyblock.*

Proof For any two vectors $z,y \in \mathbb{R}^n$ we write $u = z \wedge y$ if $u_i = \min\{z_i,y_i\}, i = 1,\ldots,n$, and $v = z \vee y$ if $v_i = \max\{z_i,y_i\}, i = 1,\ldots,n$. If V_1, V_2 are the vertex sets of two polybocks P_1, P_2, respectively, then $P_1 \cap P_2 = (\cup_{z\in V_1}[a,z]) \cap (\cup_{y\in V_2}[a,y]) = \cup_{z\in V_1,y\in V_2}[a,z] \cap [a,y] = \cup_{z\in V_1,y\in V_2}[a, z\wedge y]$. Thus, $P_1 \cap P_2$ is a polyblock of vertex set $\{z \wedge y | \, z \in V_1, y \in V_2\}$. Similarly, if V_1, V_2 are the vertex sets of two upward polyblocks Q_1, Q_2, respectively, then $Q_1 \cap Q_2 = \cup_{z\in V_1,y\in V_2}[z,b] \cap [y,b] = \cup_{z\in V_1,y\in V_2}[z \vee y, b]$, so $Q_1 \cap Q_2$ is an upward polyblock with vertex set $\{z \vee y | \, z \in V_1, y \in V_2\}$. □

Proposition 5

(1) *The maximum of an increasing function* $f(x)$ *over a polyblock is achieved at a proper vertex of this polyblock.*

(2) *The minimum of an increasing function* $f(x)$ *over an upward polyblock is achieved at a proper vertex of this upward polyblock.*

Proof We prove (1). Let $\bar{x}$ be a maximizer of $f(x)$ over a polyblock P. Since a polyblock is the downward hull of its proper vertices, there exists a proper vertex z of P

such that $\bar{x} \in [a, z]$. Then $f(z) \geq f(\bar{x})$ because $z \geq \bar{x}$, so z must be also an optimal solution. The proof of (2) is similar. □

As usual, e^i denotes the ith unit vector of $\mathbb{R}^n$, i.e., $e^i \in \mathbb{R}^n$, $e^i_i = 1$, $e^i_j = 0\ \forall j \neq i$.

Proposition 6

(1) *If $a < x < b$, then the set $[a, b] \setminus (x, b]$ is a polyblock with vertices*

$$u^i = b + (x_i - b_i)e^i, \quad i = 1, \ldots, n. \tag{3}$$

(2) *If $a < x < b$, then the set $[a, b] \setminus [a, x)$ is an upward polyblock with vertices*

$$v^i = a + (x_i - a_i)e^i, \quad i = 1, \ldots, n.$$

Proof We prove (1). Let $K_i = \{z \in [a, b] |\ x_i < z_i\}$. Since $(x, b] = \cap_{i=1,\ldots,n} K_i$, we have $[a, b] \setminus (x, b] = \cup_{i=1,\ldots,n}([a, b] \setminus K_i)$, proving the assertion because $[a, b] \setminus K_i = \{z |\ a_i \leq z_i \leq x_i,\ a_j \leq z_j \leq b_j\ \forall j \neq i\} = [a, u^i]$. The proof of (2) is similar. □

Note that $u^1, \ldots, u^n$ are the n vertices of the hyperrectangle $[x, b]$ that are adjacent to b, while $v^1, \ldots, v^n$ are the n vertices of the hyperrectangle $[a, x]$ that are adjacent to a.

3 Bilevel programming and monotonic optimization

The General Bilevel Nonlinear Programming Problem we are concerned with can be formulated as follows

$$\begin{array}{rl} & \min F(x, y) \quad \text{s.t.} \\ & g_1(x, y) \leq 0, \quad x \in \mathbb{R}^{n_1}_+, \ \ y \text{ solves} \\ (\mathrm{R}(x)) & \min \left\{ d(y) |\ g_2(C(x), y) \leq 0, \quad y \in \mathbb{R}^{n_2}_+ \right\}, \end{array} \tag{GBP}$$

where it will be assumed that

(A1) The mapping C: $\mathbb{R}^{n_1} \to \mathbb{R}^m$ and the functions $F(x, y)$, $d(y), g_1(x, y)$, $g_2(u, y)$ are continuous.

(A2) The set $D := \{(x, y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+ |\ g_1(x, y) \leq 0,\ g_2(C(x), y) \leq 0$, is nonempty and compact.

(A3) For every fixed y, $g_2(u, y)$ is a decreasing function of u; for every fixed $x \in \mathbb{R}^{n_1}_+$ the second level problem $\mathrm{R}(x)$ is solvable.

While Assumptions (A1) and (A2) are natural, Assumption (A3) may require some comments. The first proposition of Assumption (A3) means that the function $g_2(C(x), y)$ decreases when the action $C(x)$ of the leader increases (component-wise). By replacing u with $v = -u$, if necessary, this assumption also holds when $g_2(C(x), y)$ increases with $C(x)$. As for the second proposition of (A3), it is a quite common assumption in bilevel programming. In practice, one can always make it hold by adding the condition $g_2(C(x), y) \leq 0$ to the constraint on the leader, i.e., by replacing $g_1(x, y) \leq 0$ with

$$\max\{g_1(x, y), g_2(C(x), y)\} \leq 0.$$

So far the best known particular case of (GBP) is the BLP Problem:

$$\begin{aligned} &\min L(x,y) \quad \text{s.t.} \\ &A_1x + B_1y \geq c^1, \quad x \in \mathbb{R}^{n_1}_+, \ \ y \text{ solves} \\ &\min\left\{\langle d, y\rangle \mid A_2x + B_2y \geq c^2, \quad y \in \mathbb{R}^{n_2}_+\right\}, \end{aligned} \tag{BLP}$$

where $L(x,y)$ is a linear function, $d \in \mathbb{R}^{n_2}, A_i \in \mathbb{R}^{m_i\times n_1}$, $B_i \in \mathbb{R}^{m_i\times n_2}$, $c^i \in \mathbb{R}^{m_i}$, $i = 1,2$. In this case $g_1(x,y) = \max_{i=1,\dots,m_1}(c^1 - A_1x - B_1y)_i \leq 0$, $g_2(u,y) = \max_{i=1,\dots,m_2}(c^2 - u - B_2y)_i \leq 0$, while $C(x) = A_2x$.

Clearly all Assumptions (A1), (A2), and (A3) hold provided the set $D := \{(x,y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+ \mid A_1x + B_1y \geq c^1,\ A_2x + B_2y \geq c^2\}$ is a nonempty polytope (then for every fixed $x \geq 0$ satisfying $A_1x + B_1y \geq c^1$ the set $\{y \in \mathbb{R}^{n_2}_+ \mid A_2x + B_2y \geq c^2\}$ is compact, so R(x) is solvable).

Note that even (BLP) is an NP-hard nonconvex global optimization problem (see, e.g. (18)). Some other particular cases of (GBP), when the second level problem (R(x)) is convex, have been considered in (22–24). One important example of this class is the following minimum norm bilevel programming problem which appears in transportation planning applications (see, e.g. (11)):

$$\begin{aligned} &\min_x \|y(x) - c\|^2 \quad \text{s.t.} \\ &x \in X, \quad y(x) \text{ solves} \\ (\mathrm{R}(x))\ &\min\{d(y) \mid y \in \Omega(x)\}, \end{aligned}$$

where X and $\Omega(x)$, for every fixed x, are polytopes in $\mathbb{R}^n_+$, while $d(y)$ is a convex function.

We now show that (GBP) is in fact a monotonic optimization problem.

In view of Assumptions (A1) and (A2) the set $\{(C(x), d(y)) \mid (x,y) \in D\}$ is compact, so without loss of generality we can assume that this set is contained in some box $[a,b] \subset \mathbb{R}^{m+1}_+$:

$$a \leq (C(x), d(y)) \leq b \quad \forall (x,y) \in D. \tag{4}$$

Let

$$U = \{u \in \mathbb{R}^m \mid (\exists (x,y) \in D)\ u \geq C(x)\}, \tag{5}$$

$$W = \{(u,t) \in \mathbb{R}^m \times \mathbb{R} \mid\ \exists (x,y) \in D,\ u \geq C(x),\ t \geq d(y)\} \tag{6}$$

and define

$$\theta(u) = \min\{d(y) \mid (x,y) \in D,\ u \geq C(x)\}, \tag{7}$$

$$f(u,t) = \min\{F(x,y) \mid (x,y) \in D, u \geq C(x),\ t \geq d(y)\}. \tag{8}$$

Proposition 7

(1) *The function $\theta(u)$ is finite for every $u \in U$ and satisfies: $\theta(u') \leq \theta(u)$ whenever $u' \geq u \in U$.*
(2) *The function $f(u,t)$ is finite on the set W. If $(u',t') \geq (u,t) \in W$ then $(u',t') \in W$ and $f(u',t') \leq f(u,t)$.*
(3) *$u \in U$ for every $(u,t) \in W$.*

Proof (1) If $u \in U$ then there exists $(x,y) \in D$ such that $C(x) \leq u$, hence $\theta(u) < +\infty$. It is also obvious that $u' \geq u \in U$ implies $\theta(u') \leq \theta(u)$.

(2) By Assumption (A2), the set D of all $(x,y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+$ satisfying $g_1(x,y) \leq 0,\ g_2(C(x),y) \leq 0$ is nonempty, compact. Then for every $(u,t) \in W$ the feasible set of the problem determining $f(u,t)$ is nonempty, compact, hence $f(u,t) < +\infty$ because $F(x,y)$ is continuous by (A1). Furthermore, if $(u',t') \geq (u,t) \in W$ then obviously $(u',t') \in W$, and $f(u',t') \leq f(u,t)$.

(3) Obvious. □

In addition to (A1), (A2), and (A3), we now make a further assumption:
(A4) The function $\theta(u)$ is continuous on int$U$, while $f(u,t)$ is continuous on int$W$.

Proposition 8 *If all the functions* $F(x,y)$, $g_1(x,y)$, $g_2(C(x),y)$, $d(y)$ *are convex then Assumption* (A4) *holds, with* $\theta(u), f(u,t)$ *being convex functions.*

Proof Observe that if $\theta(u) = d(y)$ for x satisfying $g_2(C(x),y) \leq 0, C(x) \leq u$, and $\theta(u') = d(y')$ for x' satisfying $g_2(C(x'),y') \leq 0, C(x') \leq u'$, then $g_2(C(\alpha x + (1-\alpha)x'), \alpha y + (1-\alpha)y') \leq \alpha g_2(C(x),y) + (1-\alpha)g_2(C(x'),y')$, and $C(\alpha x + (1-\alpha)x') \leq \alpha C(x) + (1-\alpha)C(x') \leq \alpha u + (1-\alpha)u'$, so that $(\alpha x + (1-\alpha)x', \alpha y + (1-\alpha)y')$ is feasible to the problem determining $\theta(\alpha u + (1-\alpha)u')$, hence $\theta(\alpha u + (1-\alpha)u') \leq d(\alpha y + (1-\alpha)y') \leq \alpha d(y) + (1-\alpha)d(y') = \alpha\theta(u) + (1-\alpha)\theta(u')$. Therefore, the function $\theta(u)$ is convex, and hence continuous on int$U$. The convexity and hence, the continuity of $f(u,t)$ on int$W$ is proved similarly. □

Corollary 1 *The set*

$$G = \{(u,t) \in [a,b] |\ t - \theta(u) \leq 0\}$$

is downward and closed, while $\partial G^+ \subset \{(u,t) \in [a,b] |\ t - \theta(u) = 0\}$.

Proof The function $(u,t) \mapsto t - \theta(u)$ is increasing on the box $[a,b]$ since it is increasing on $W \cap [a,b]$ and equals $-\infty$ for $(u,t) \in [a,b] \setminus W$. (Note that always $b \in W$). □

Proposition 9 *The problem* (GBP) *is equivalent to the monotonic optimization problem*

$$\min\{f(u,t) |\ (u,t) \in G\} \tag{Q}$$

in the sense that min (GBP)= min(Q) *and if* $(\bar{u},\bar{t})$ *solves (Q) then any optimal solution* $(\bar{x},\bar{y})$ *of the problem*

$$\min\{F(x,y) |\ g_1(x,y) \leq 0, g_2(\bar{u},y) \leq 0, \bar{u} \geq C(x), \bar{t} \geq d(y), x \geq 0, y \geq 0\}$$

solves (GBP).

Proof If (x,y) is feasible to (GBP) then $g_1(x,y) \leq 0,\ g_2(C(x),y) \leq 0$ and taking $u = C(x),\ t = d(y) = \theta(C(x))$ we have $(u,t) \in G$, hence

$$\begin{aligned} F(x,y) \geq \min\big\{&F(x',y') |\ g_1(x',y') \leq 0,\ g_2(C(x'),y') \leq 0,\\ &u \geq C(x'),\ t \geq d(y'),\ x' \geq 0, y' \geq 0\big\}\\ = f(u,t) &\geq \min\{f(u',t') |\ (u',t') \in G\} = \min\,(Q). \end{aligned}$$

Conversely, if $(u,t) \in G$, i.e., $t \le \theta(u)$, then the inequalities $u \ge C(x),\ t \ge d(y)$ imply $d(y) \le t \le \theta(u) \le \theta(C(x))$, hence

$$\begin{aligned} f(u,t) &= \min\{F(x,y)|\ g_1(x,y) \le 0,\ g_2(C(x),y) \le 0, \\ &\qquad u \ge C(x),\ t \ge d(y),\ x \ge 0,\ y \ge 0\} \\ &\ge \min\{F(x,y)|\ g_1(x,y) \le 0,\ g_2(C(x),y) \le 0,\ \theta(C(x)) \ge d(y)\} \\ &= \min(\text{GBP}). \end{aligned}$$

Consequently, min (GBP) = min (Q). The last assertion of the proposition obviously follows. □

Remark 1 As seen from the above proof, (GBP) can be reformulated as

$$\min\{F(x,y)|\ g_1(x,y) \le 0,\ g_2(C(x),y) \le 0,\ \theta(C(x)) - d(y) \ge 0\},$$

where the function $(x,y) \mapsto \theta(C(x)) - d(y)$ is the composition of the decreasing function $(u,t) \mapsto \theta(u) - t$ with $u = C(x), t = d(y)$. Therefore, the constraint $\theta(C(x)) - d(y) \ge 0$ is a composite monotonic constraint of the type considered in (37), and (GBP) can be solved by the method developed in that paper.

4 Polyblock approximation

For convenience of notation let us set

$$z = (u,t) \in \mathbb{R}^{m+1}, \qquad \varphi(z) = t - \theta(u). \tag{9}$$

We then have a continuous increasing function $\varphi(z)$ such that $G = \{z \in [a,b]|\ \varphi(z) \le 0\}$, and the problem (Q) can be rewritten as

$$\min\{f(z)|\ z \in G\}. \tag{Q}$$

For solving this monotonic optimization problem one possible method is to use *polyblock approximation*, as developed in (32).

If $\varphi(a) > 0$, i.e. $a \notin G$ then $G \cap [a,b] = \emptyset$ and the problem is infeasible. On the other hand, if $\varphi(b) := b_{m+1} - \theta(b_1, \ldots, b_m) = 0$ then $b \in G$, and since $f(z)$ is decreasing b is a global optimal solution. Barring these trivial cases, we can thus assume

$$\varphi(a) \le 0 < \varphi(b). \tag{10}$$

Now to solve (Q) by polyblock approximation we start from the initial polyblock $P_1 = [a,b]$.

Since the polyblock $P_1 \supset G$ is an outer approximation of the feasible set $G \subset [a,b]$, the minimum of the objective function $f(z)$ over P_1 gives a lower bound for the optimal value of (Q). But, since $f(z)$ is decreasing, its minimum over P_1 must be achieved at some proper vertex of P_1. Thus, if V_1 denotes the vertex set of P_1 and

$$v^1 \in \operatorname{argmin}\{f(z)|\ z \in V_1\},$$

then $f(v^1) \le \min\{f(x)|\ x \in G\}$. If it so happens that $v^1 \in G$ then v^1 solves problem (Q). On the other hand, if $v^1 \notin G$ then we construct as follows a new polyblock, smaller than P_1, excluding v^1 but still containing G. Let $w^1 = \pi_a(v^1)$ (see (1)), i.e. $w^1 = v^1 - \alpha_1(v^1 - a)$ where

$$\alpha_1 = \min\{\alpha |\ v^1 - \alpha(v^1 - a) \in G\}. \tag{11}$$

Then $w^1 \in \partial^+ G$, so no point of G can be found in the cone $K_1 = \{z \in \mathbb{R}^{m+1}_+ |\ z > w^1\}$. Noting that $P_1 \supset G$, we infer that $P_1 \backslash K_1 \supset G$, i.e. the polyblock $P_2 = P_1 \backslash K_1 = P_1 \cap ([a, b] \setminus K_1)$ still contains all feasible points of interest, in particular, it still contains at least a global optimal solution. To compute the proper vertex set V_2 of P_2 we use the following proposition established in (33).

For any two vectors $v, v' \in V_1$ define $J(v, v') := \{j |\ v_j > v'_j\}$.

Proposition 10 *Let V_1 be the proper vertex set of P_1, and $V_1^* = \{v \in V_1 |\ v > w^1\}$. Then the vertex set of the polyblock $P_2 = P_1 \setminus K_1$ is*

$$V_2' = (V_1 \setminus V_1^*) \cup \{v^i = v + (w_i^1 - v_i)e^i |\ i = 1, \ldots, m + 1,\ v \in V_1^*\}. \tag{12}$$

The proper vertex set V_2 of P_2 is obtained by removing from V_2' all those $v^i = v + (w_i^1 - v_i)e^i$ with $v \in V_1^*$, such that $J(v, v') = \{i\}$ for some $v' \in V_1$ satisfying $v' \geq w^1$.

Proof See (33). □

Thus, while loosing no better solution than the current best feasible solution, the polyblock P_2 with proper vertex set V_2 is smaller than P_1 because it has excluded v^1 along with all points in $P_2 \cap K_1$. At this point observe that if we already know a feasible solution $\bar{v}^1$ then any vertex $v \in V_2$ such that $f(v) \geq f(\bar{v}^1)$ can be removed, since, $f(.)$ being decreasing, no better solution than $\bar{v}^1$ can be found in $[v, b]$. To simplify the notation, denote the set that remains from V_2 after all this pruning operation again by V_2 and the corresponding polyblock again by P_2. Since P_2 still contains at least a global optimal solution, the above procedure can be repeated with P_2 in place of P_1, and so on. Continuing this way we will generate a nested sequence of polyblocks $P_1 \supset P_2 \supset \cdots$, containing G (or a subset of G including at least a global optimal solution) and approximating this set more and more closely. It can be proved that if

$$(\exists \rho > 0)\ (\forall z \notin G) \quad \|z - a\| > \rho$$

(a mild condition which can easily be made to hold) then, whenever the procedure is infinite we have

$$f(v^k) := \min\{f(v) |\ v \in P_k\} \nearrow \min\{f(z) |\ z \in G\}.$$

On the other hand, if at some iteration k it so happens that $v^k = w^k$ then v^k is feasible, hence optimal; also, if $V_k = \emptyset$ this means that no better feasible solution than the current incumbent exists, hence the latter is optimal.

5 Branch-reduce-and-bound method

Like most outer approximation procedures, the above polyblock approximation algorithm is easily implementable but practical only for problems (Q) with small values of m, typically $m \leq 5$. For problems with larger values of m a method of branch and cut type, called BRB method, that has been developed in (32) and refined in (36), is more suitable. We next describe this method.

As the name indicates, the BRB method proceeds according to the standard branch and bound scheme with three basic operations: branching, reducing (the partition sets) and bounding.

(1) Branching consists in a successive rectangular partition of the initial box $M_0 = [a,b] \subset \mathbb{R}^m \times \mathbb{R}$, following a chosen subdivision rule. As will be explained shortly, branching should be performed upon the variables $u \in \mathbb{R}^m$, so the subdivision rule is induced by a standard bisection upon these variables.
(2) Reducing consists in applying valid cuts to reduce the size of the current partition set $M = [p,q] \subset [a,b]$. The box $[p',q']$ obtained from M as a result of the cuts is referred to as a *valid reduction* of M.
(3) Bounding consists in estimating a *valid lower bound* $\beta(M)$ for the objective function value $f(z)$ over the feasible portion contained in the valid reduction $[p',q']$ of a given partition set $M = [p,q]$.

Throughout the sequel, for any vector $z = (u,t) \in \mathbb{R}^m \times \mathbb{R}$, we define $\hat{z} = u$.

5.1 Branching

Since an optimal solution (u,t) of the problem (Q) must satisfy $t = \theta(u)$, it suffices to determine the values of the variables u in an optimal solution. This suggests that instead of branching upon $z = (u,t)$ as usual, we should branch upon u, according to the standard bisection rule. Specifically, given a box $M = [p,q]$, we select $r \in \{1,\dots,m\}$ such that $q_r - p_r = \max\{q_i - p_i |\ i = 1,\dots,m\}$, and bisect M by the hyperplane $u_r = \frac{1}{2}(p_r + q_r)$, thus generating two subboxes:

$$\left[p,\ q - \frac{1}{2}(q_r - p_r)e^r\right], \qquad \left[p + \frac{1}{2}(q_r - p_r)e^r,\ q\right].$$

As is well known (see, e.g. (28)) this subdivision method has the important property that for any infinite nested sequence $M_k = [p^k, q^k], k = 1,2,\dots$, such that each M_{k+1} is one of the two subboxes of M_k via the bisection we have $\lim_{k\to+\infty}(\hat{q}^k - \hat{p}^k) = 0$.

5.2 Valid reduction

At a given stage of the BRB algorithm for (Q), a feasible point $\bar{z} \in G$ is available which is the best so far known. Let $\gamma = f(\bar{z})$ and let $[p,q] \subset [a,b]$ be a box generated during the partitioning procedure which is still of interest. Since an optimal solution of (Q) is attained at a point satisfying $\varphi(z) := t - \theta(u) = 0$, the search for a feasible solution of (Q) in $[p,q]$ such that $f(z) \le \gamma$ can be restricted to the set $B_\gamma \cap [p,q]$, where

$$B_\gamma := \{z |\ f(z) - \gamma \le 0,\ \varphi(z) \le 0 \le \varphi(z)\}. \tag{13}$$

The reduction operation aims at replacing the box $[p,q]$ with a smaller box $[p',q'] \subset [p,q]$ without losing any point $y \in B_\gamma \cap [p,q]$, i.e., such that

$$B_\gamma \cap [p',q'] = B_\gamma \cap [p,q].$$

The box $[p',q']$ satisfying this condition is referred to as a γ*-valid reduction* of $[p,q]$ and denoted by $\mathrm{red}_\gamma[p,q]$.

First of all, since the points (u,t) of interest in $[p,q]$ must satisfy $t = \theta(u)$, hence $\theta(\hat{q}) \le t \le \theta(\hat{p})$, by making the substitution $\theta(\hat{q}) \leftarrow p_{m+1}, \theta(\hat{p}) \leftarrow q_{m+1}$ wherever necessary, we can assume that

$$p_{m+1} \geq \theta(\hat{q}), \qquad q_{m+1} \leq \theta(\hat{p}). \tag{14}$$

Recall that e^i denotes the ith unit vector of $\mathbb{R}^{m+1}$, i.e., $e^i \in \mathbb{R}^{m+1}$, $e^i_i = 1$, $e^i_j = 0 \ \forall j \neq i$.

Proposition 11

(1) *If* $\varphi(q) < 0$ *or* $f(q) - \gamma > 0$, *then* $B_\gamma \cap [p,q] = \emptyset$, *i.e.,* $\mathrm{red}_\gamma[p,q] = \emptyset$.
(2) *If* $\varphi(q) \geq 0$ *and* $f(q) \leq \gamma$, *then* $\mathrm{red}_\gamma[p,q] = [p',q']$ *where*

$$q' = p + \sum_{i=1}^{m+1} \beta_i(q_i - p_i)e^i \quad \textit{with} \tag{15}$$

$$\beta_i = \max\left\{\beta |\ 0 < \beta \leq 1,\ \varphi\left(p + \beta\left(q_i - p_i\right)e^i\right) \leq 0\right\}, \tag{16}$$

$$p' = q' - \sum_{i=1}^{m+1} \alpha_i(q'_i - p_i)e^i \quad \textit{with} \tag{17}$$

$$\begin{aligned}\alpha_i = \max\{\alpha |\ 0 < \alpha \leq 1,\ &\varphi(q' - \alpha(q'_i - p_i)e^i) \geq 0,\\ &f(q' - \alpha(q'_i - p_i)e^i) \leq \gamma\}.\end{aligned} \tag{18}$$

Proof

(1) If $\tilde{f}(q) := \min\{\gamma - f(q),\ \varphi(q)\} < 0$, then, since $\tilde{f}(z)$ is increasing, $\tilde{f}(z) \leq \tilde{f}(q) < 0$ for every $z \in [p,q]$. Similarly, if $\varphi(p) > 0$, then $\varphi(z) > 0$ for every $z \in [p,q]$. In both cases, $B_\gamma \cap [p,q] = \emptyset$.
(2) Let $z \in [p,q]$ satisfy $f(z) \leq \gamma$ and $\varphi(z) = 0$. If $z \not\leq q'$ then there is $i \in \{1,\ldots,m+1\}$ such that $z_i > q'_i = p_i + \beta_i(q_i - p_i)$, i.e., $z_i = p_i + \beta(q_i - p_i)$ with $\beta > \beta_i$ and from (16) it follows that $\varphi(p + \beta(q_i - p_i)e^i) > 0$, which implies that $\varphi(z) > 0$, conflicting with $z \in B_\gamma$. Similarly, if $z \not\geq p'$ then there is $i \in \{1,\ldots,m+1\}$ such that $z_i < p'_i = q'_i - \alpha_i(q'_i - p_i)$, i.e., $z_i = q'_i - \alpha(q'_i - p_i)$ with $\alpha > \alpha_i$. In view of (18), this implies that either $\varphi(z) \leq \varphi(q' - \alpha(q'_i - p_i)e^i) < 0$, conflicting with $\varphi(z) = 0$, or that $f(z) \geq f(q' - \alpha(q'_i - p_i)e^i) > \gamma$, conflicting with $f(z) \leq \gamma$. Thus, any $z \in [p,q]$ such that $f(z) \leq \gamma$, $\varphi(z) = 0$ must satisfy $p' \leq z \leq q'$, as was to be proved. □

Remark 2 It can easily be verified that the box $[p',q'] = \mathrm{red}_\gamma[p,q]$ still satisfies $\varphi(q') \geq 0$, $f(q') \leq \gamma$.

5.3 Valid bounds

Let $M := [p,q]$, be a partition set which is supposed to have been reduced, so that according to Remark 2:

$$\varphi(q) \geq 0, \quad f(q) \leq \gamma.$$

Let us now examine how to compute a lower bound $\beta(M)$ for

$$\min\{f(z) |\ z \in [p,q],\ \varphi(z) = 0\}.$$

Since $f(z)$ is decreasing, an obvious lower bound is $f(q)$. We will shortly see that to ensure convergence of the BRB Algorithm, it suffices that the lower bounds satisfy

$$\beta(M) \geq f(q). \tag{19}$$

We shall refer to such a bound as a *valid lower bound.*

As in the proof of Proposition 11 let $\tilde{f}(z) = \min\{\gamma - f(z),\ \varphi(z)\}$. Define

$$G = \{z \in [p,q] \mid \varphi(z) \le 0\}, \qquad \tilde{H} = \{z \in [p,q] \mid \tilde{f}(z) \ge 0\}. \tag{20}$$

If $\varphi(q) = 0 \le \tilde{f}(q)$, then obviously q is an exact minimizer of $f(z)$ over the feasible points in $[p,q]$ at least as good as the current best, and can be used to update the current best solution.

Suppose now that $\varphi(p) < 0,\ \tilde{f}(q) \ge 0$, i.e., $p \in G,\ q \in \tilde{H}$.

For each $z \in [p,q]$ such that $z \in \tilde{H} \setminus G$ let $\pi_p(z)$ be, as previously, the first point where the line segment from z to p meets the upper boundary of G; that is, let

$$\begin{aligned} \pi_p(z) &= z - \bar{\alpha}(z-p) \quad \text{with} \\ \bar{\alpha} &= \min\{\alpha \mid \varphi(z - \alpha(z-p)) \le 0\}. \end{aligned} \tag{21}$$

Obviously, $\varphi(\pi_p(z)) = 0$.

Lemma 1 *If $v = \pi_p(q),\ v^i = q - (q_i - v_i)e^i,\ i = 1,\dots,m+1$, and $I = \{i \mid v^i \in \tilde{H}\}$, then a valid lower bound over $M = [p,q]$ is*

$$\begin{aligned} \beta(M) &= \min\{f(v^i) \mid i \in I\} \\ &= \min\{f(u,t) \mid g_1(x,y) \le 0, \\ &\qquad g_2(\hat{u}_i, y) \le 0,\ (C(x), d(y)) \le v^i\ (i \in I),\ x \ge 0,\ y \ge 0\}. \end{aligned}$$

Proof This follows since the polyblock with vertices $v^i, i \in I$, contains all feasible $z = (u,t)$ still of interest in $[p,q]$. □

Remark 3 Each box $M_i := [v^i, q]$ can be reduced by the method presented above. If $[p'^i, q'^i] = \mathrm{red}[v^i, q],\ i = 1,\dots,m+1$, then without much extra effort, we can have a more refined lower bound, namely

$$\beta(M) = \min_{i \in I} f(q'^i), \qquad I = \{i \mid v^i \in \tilde{H}\}.$$

The above procedure amounts to constructing a polyblock $Z \supset B_\gamma \cap [p,q] = G \cap \tilde{H}$, which is possible because G is a downward and $\tilde{H}$ an upward set. To have a tighter lower bound, one can even construct a sequence of polyblocks $Z_1 \supset Z_2, \dots$, approximating the set $B_\gamma \cap [p,q]$ more and more closely. In principle, with the polyblock approximation procedure described in Sect. 3, it is possible to obtain a bound as tight as desired. Since, however, the computation cost increases rapidly with the accuracy requirement, a trade-off must be made, so practically just one approximating polyblock as in the above Lemma is used.

5.4 Algorithm and convergence

We are now in a position to state the proposed algorithm for (Q).

Recall that the problem is $\min\{f(z) \mid z \in [a,b],\ \varphi(z) = 0\}$ and we assume (A1)–(A4), so always $b \in W, \hat{b} = (b_1,\dots,b_m) \in U$, and $(\hat{b}, \theta(\hat{b}))$ yields a feasible solution.

Basic BRB algorithm for (Q)

Step 0 Start with $\mathcal{P}_1 = \{M_1\}, M_1 = [a,b], \mathcal{R}_1 = \emptyset$. Let CBS be the best feasible solution available and CBV (current best value) the corresponding value of $f(z)$. (At least CBV $= f(\hat{b}, \theta(\hat{b}))$). Set $k = 1$.

Step 1 For each box $M \in \mathcal{P}_k$ and for $\gamma = \text{CBV}$:

(1) Compute the γ-valid reduction $\text{red}_\gamma M$ of M.
(2) Delete M if $\text{red}_\gamma M = \emptyset$.
(3) Replace M by $\text{red}_\gamma M$ if $\text{red}_\gamma M \neq \emptyset$.
(4) If $\text{red}_\gamma M = [p, q]$ then compute a valid lower bound $\beta(M)$ for $f(z)$ over the feasible solutions in M.

Step 2 Let $\mathcal{P}'_k$ be the collection of boxes that results from $\mathcal{P}_k$ after completion of Step 1. Update CBS and CBV. From $\mathcal{R}_k$ remove all $M \in \mathcal{R}_k$ such that $\beta(M) \geq \text{CBV}$ and let $\mathcal{R}'_k$ be the resulting collection. Let $\mathcal{M}_k = \mathcal{R}'_k \cup \mathcal{P}'_k$.

Step 3 If $\mathcal{M}_k = \emptyset$ then terminate: CBV is the optimal value and CBS (the feasible solution $\bar{z}$ with $f(\bar{z}) = \text{CBV}$) is an optimal solution.

Step 4 If $\mathcal{M}_k \neq \emptyset$, select $M_k \in \text{argmin}\{\beta(M) |\, M \in \mathcal{M}_k\}$. Divide M_k into two subboxes by the standard bisection. Let $\mathcal{P}_{k+1}$ be the collection of these two subboxes of M_k.

Step 5 Let $\mathcal{R}_{k+1} = \mathcal{M}_k \setminus \{M_k\}$. Increment k and return to Step 1.

Proposition 12 *Whenever infinite, the* BRB *Algorithm generates an infinite filter of boxes* $\{M_{k_l}\}$ *whose intersection yields a global optimal solution.*

Proof If the algorithm is infinite, it generates at least an infinite nested sequence of boxes $M_{k_\nu}, \nu = 1, 2, \ldots$ For simplicity we write M_ν for M_{k_ν}. Let $M_\nu = [p^\nu, q^\nu]$. The subdivision method ensures that $\cap_{\nu=1}^{+\infty}[\hat{p^\nu}, \hat{q^\nu}] = \bar{u} \in \mathbb{R}^m$. Furthermore, it can be assumed that $\theta(\hat{q^\nu}) \leq p^\nu_{m+1} \leq q^\nu_{m+1} \leq \theta(\hat{p^\nu})$ (see (14)), so $\lim_{\nu\to+\infty} p^\nu_{m+1} = \theta(\bar{u})$, and $\cap_{l=1}^{\infty} M_{k_l} = \{z^*\}$. Since $M_{k_l} = [p^{k_l}, q^{k_l}]$ with $\varphi(p^{k_l}) \leq 0 \leq \varphi(q^{k_l})$, and $z^* = \lim p^{k_l} = \lim q^{k_l}$, it follows that $\varphi(z^*) \leq 0 \leq \varphi(z^*)$, i.e., z^* is a feasible solution. Furthermore,

$$f(p^{k_l}) \geq \beta(M_{k_l}) \geq f(q^{k_l}),$$

whence $\lim_{l\to+\infty} \beta(M_{k_l}) = f(z^*)$. On the other hand, since M_{k_l} corresponds to the minimum of $\beta(M)$ among the current set of boxes, we have $\beta(M_{k_l}) \leq \min\{f(z) |\, z \in G \cap [a, b]\}$ and, consequently,

$$f(z^*) \leq \min\{f(z) |\, z \in G \cap [a, b]\},$$

proving that z^* is an optimal solution. □

6 Implementation issues

In this section, we discuss the main operations involved in the selection of the initial box, the reduction and the bounding operations. Note that when the functions $F(x, y), g_1(x, y), g_2(u, y), d(y)$ are convex, and $C(x)$ is affine, the subproblems of computing $\theta(u), f(u, t)$ are convex and can be solved by efficient algorithms.

6.1 Initial box

From (4) and (14) the initial box $[a, b] \subset \mathbb{R}^{m+1}$ can be determined as follows:

$$a_i = \min\{C_i(x) |\, (x, y) \in D\}\ (i = 1, \ldots, m), \quad a_{m+1} = \theta(\hat{b}),$$
$$b_i = \max\{C_i(x) |\, (x, y) \in D\}\ (i = 1, \ldots, m), \quad b_{m+1} = \theta(\hat{a}).$$

where $\hat{a} = (a_1, \dots, a_m)$, $\hat{b} = (b_1, \dots, b_m)$,

$$D = \{(x, y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+ |\ g_1(x, y) \le 0, g_2(C(x), y) \le 0\}.$$

6.2 Reduction operation

To simplify certain formulations it is convenient to write $s = m+1$ and $\hat{a} = (a_1, \dots, a_m)$ for any $a = (a_1, \dots, a_m, a_s) \in \mathbb{R}^s$. In each iteration of Polyblock approximation, the following operation is needed (see (11), (21)): given a point $z = (u, t) \in [a, b] \setminus G$ compute the point $\pi_a(z) = z - \bar{\alpha}(z - a)$ such that $z - \bar{\alpha}(z - a) \in \partial^+ G$. Setting $t(\alpha) = t - \alpha(t - a_s)$, $u(\alpha) = u - \alpha(u - \hat{a})$ one has

$$\bar{\alpha} = \max\{\alpha |\ t(\alpha) - \theta(u(\alpha)) \ge 0\}.$$

Since $t(\alpha) - \theta(u(\alpha))$ is a decreasing function of one variable $\alpha \in [0, 1]$, one way to compute $\bar{\alpha}$ is to proceed by Bolzano bisection. However since $\theta(u + \alpha(u - \hat{a}))$ is the optimal value of an optimization problem (see (7)), this method would require solving the latter problem for some sequence of values of α. A more efficient way to compute $\bar{\alpha}$ is furnished by the next proposition.

Proposition 13 *For any $z = (u, t) \in [a, b] \setminus G$ the number $\bar{\alpha}$ such that $\pi_a(z) = z - \bar{\alpha}(z - a)$ is equal to the optimal value of the program*

$$\begin{array}{ll} & \max_{x,y,\alpha}\ \alpha \\ s.t. & \left| \begin{array}{l} d(y) \le t(\alpha), \quad C(x) \le u(\alpha), \\ g_2(C(x), y) \le 0, \\ x \ge 0, \quad y \ge 0, \quad 0 \le \alpha \le 1. \end{array} \right. \end{array} \tag{22}$$

Proof Denote the optimal value of (22) by $\tilde{\alpha}$. If $t(\alpha) - \theta(u(\alpha)) \ge 0$, then for $x \in \mathbb{R}^{n_1}_+$, $y \in \mathbb{R}^{n_2}_+$ such that $g_2(C(x), y) \le 0$, $C(x) \le u(\alpha)$, $d(y) = \theta(u(\alpha))$ one has $t(\alpha) - d(y) = t(\alpha) - \theta(u(\alpha)) \ge 0$, i.e., (x, y, α) satisfies the constraints of (22). Hence $\bar{\alpha} \le \tilde{\alpha}$. On the other hand, if $(\tilde{x}, \tilde{y}, \tilde{\alpha})$ solves (22) then $d(\tilde{y}) \le t(\tilde{\alpha})$, $C(\tilde{x}) \le u(\tilde{\alpha}), g_2(C(\tilde{x}), \tilde{y}) \le 0$, hence $t(\tilde{\alpha}) - \theta(u(\tilde{\alpha})) \ge t(\tilde{\alpha}) - d(\tilde{y}) \ge 0$. This shows that $\tilde{\alpha} \le \bar{\alpha}$, and hence, $\tilde{\alpha} = \bar{\alpha}$. □

Similarly, each of the numbers α_i defined in (18) can be computed by solving a suitable single optimization problem. Also observe that to have a valid reduction of $M = [p, q]$ we need only know how to compute α_i, β_i for $i = 1, \dots, m$. In fact, when these numbers have been computed, then a valid reduction of $M = [p, q]$ can be taken to be $\text{red}_\gamma[p, q] = [p', q']$, with

$$\hat{q}' = \hat{p} + \sum_{i=1}^{m} \beta_i (q_i - p_i) e^i, \qquad q'_s = \min\{q_s, \theta_\gamma(\hat{p}')\}, \tag{23}$$

$$\hat{p}' = \hat{q}' - \sum_{i=1}^{m} \alpha_i (q'_i - p_i) e^i, \qquad p'_s = \theta_\gamma(\hat{q}'), \tag{24}$$

where for every $u \in [p, q]$:

$$\begin{aligned} \theta_\gamma(u) = \min\{d(y) |\ & g_2(u, y) \le 0,\ u \ge C(x), \\ & F(x, y) \le \gamma,\ x \in \mathbb{R}^{n_1}_+, y \in \mathbb{R}^{n_2}_+\} \end{aligned} \tag{25}$$

(note that $\theta_\gamma(u) \ge \theta(u)$, so $\varphi(q') \ge \varphi_\gamma(q') := q'_s - \theta_\gamma(\hat{q}') \ge 0, f(q') \ge \gamma$).

Proposition 14 *For $i = 1,\ldots,m$ we have*

$$\alpha_i = \max_{x,y,\alpha} \alpha \quad \text{s.t.} \left| \begin{array}{l} F(x,y) \le \gamma \\ g_1(x,y) \le 0, \quad g_2(C(x),y) \le 0 \\ \hat{q}' - \alpha(q'_i - p_i)\hat{e}^i \ge C(x), \quad q'_s \ge d(y), \\ x \ge 0, y \ge 0, \quad 0 \le \alpha \le 1. \end{array} \right. \tag{26}$$

Proof Let $u^i(\alpha) = \hat{q}' - \alpha(q'_i - p_i)\hat{e}^i$. If (x,y,α) satisfies (26), then clearly $f(u^i(\alpha), q'_s) \le \gamma$. Furthermore, $q'_s \ge d(y) \ge \theta_\gamma(C(x)) \ge \theta_\gamma(u^i(\alpha))$, hence $\varphi_\gamma(q' - \alpha(q'_i - p_i)e^i) \ge 0$.

Conversely, if $0 \le \alpha \le 1$ and $\varphi_\gamma(q' - \alpha(q'_i - p_i)e^i) \ge 0$, $f(q' - \alpha(q'_i - p_i)e^i) \le \gamma$, i.e., $q'_s - \theta_\gamma(u^i(\alpha)) \ge 0$, $f(u^i(\alpha), q'_s) \le \gamma$, then for $x, y \ge 0$ such that $g_1(x,y) \le 0$, $g_2(C(x),y) \le 0$, $C(x) \le u^i(\alpha)$, $d(y) \le q'_s$, and $F(x,y) = f(u^i(\alpha), q'_s)$ we have $F(x,y) \le \gamma$, hence (x,y,α) satisfies (26). □

The computation of the exact value of β_i, $i = 1,\ldots,m$, in (18) is more involved, so instead of β_i we compute an approximate value $\bar{\beta}_i$ such that $\bar{\beta}_i \ge \beta_i$. The latter condition ensures that the box $[p', q']$ obtained that way will still be a valid reduction of $[p,q]$, while $\bar{\beta}_i$ can be computed by solving a single program, as shown by the following proposition.

Proposition 15 *Let $\varepsilon > 0$. For $i = 1,\ldots,m$ we have*

$$\beta_i \le \bar{\beta}_i = \min_{x,y,\beta} \beta \quad \text{s.t.} \left| \begin{array}{l} g_2(C(x),y) \le 0, \quad C(x) \le \hat{p} + \beta(q_i - p_i)\hat{e}^i, \\ p_s \ge d(y) + \varepsilon, \\ x \ge 0, \ y \ge 0, \quad 0 \le \beta \le 1. \end{array} \right. \tag{27}$$

Furthermore, $\bar{\beta}_i \to \beta_i$ as $\varepsilon \searrow 0$.

Proof Let $(\bar{x}, \bar{y}, \bar{\beta}_i)$ solve the problem (27). If $\bar{\beta}_i = 1$ then obviously $\beta_i \le \bar{\beta}_i$. Otherwise, letting $u^i(\beta_i) = \hat{p} + \beta(q_i - p_i)\hat{e}^i$ we show that $p_s - \theta_\gamma(u^i(\bar{\beta}_i)) = \varepsilon$. Indeed, since $\theta_\gamma(u)$ is decreasing, if $p_s - \theta_\gamma(u^i(\bar{\beta}_i)) > \varepsilon$, then with $\beta < \bar{\beta}_i$ one could still have $p_s - \theta_\gamma(u^i(\beta)) \ge \varepsilon$, i.e., there would exist $x \in \mathbb{R}^{n_1}_+, y \in \mathbb{R}^{n_2}_+$ satisfying $F(x,y) \le \gamma$, $g_2(C(x),y) \le 0$, $C(x) \le u^i(\beta)$, $p_s \ge d(y) + \varepsilon$, conflicting with $\bar{\beta}_i$ being the smallest of all such β. Since $\varphi_\gamma(u^i(\beta_i), p_s) \le 0 < \varepsilon = \varphi_\gamma(u^i(\bar{\beta}_i), p_s)$, while the function $\beta \mapsto \varphi_\gamma(u^i(\beta), p_s)$ is increasing it follows that $\beta_i \le \bar{\beta}_i$. Furthermore, as $\varepsilon \to 0$, if $\bar{\beta}_i \to \beta'_i > \beta_i$ then by continuity $\varphi_\gamma(u^i(\bar{\beta}_i), p_s) \to \varphi_\gamma(u^i(\beta'_i), p_s) > 0$, conflicting with $\varphi_\gamma(u^i(\bar{\beta}_i), p_s) = \varepsilon \to 0$. Therefore, $\bar{\beta}_i \to \beta_i$. □

Remark 4 If $\varphi_\gamma(p_i) = 0$ then obviously $\beta_i = 1$ so $\bar{\beta}_i = 0$. In practice, in the computation of $\bar{\beta}_i$ one can take, e.g., $\varepsilon = 0.01(q_i - p_i)$.

For Bilevel Convex Programs where $F(x,y)$, $g_1(x,y), g_2(u,y), d(y)$ are convex and $C(x)$ is affine (so that Assumptions (A1) through (A4) are satisfied), the above subproblems are convex programs. In the case of BLP these are linear programs, with

$$\begin{aligned} &F(x,y) = L(x,y), \qquad d(y) = \langle d, y\rangle, \qquad C(x) = A_2(x), \\ &g_1(x,y) = -A_1x - B_1y + c^1, \qquad g_2(u,y) = -u - B_2y + c^2, \\ &D = \{(x,y) |\, A_ix + B_iy \ge c^i \ (i = 1,2), \quad x \ge 0, \ y \ge 0\}. \end{aligned}$$

6.3 Lower bounding

By Proposition 12, the Basic BRB Algorithm converges provided the bounds are valid, i.e., satisfy (19):

$$\beta(M) \geq f(q).$$

The convergence speed depends, however, on the quality of the bounds and in most cases, much better bounds can be computed by exploiting specific structures underlying the problem. In Sect. 5.3 we already indicated a general bounding method based on polyblock approximation. For problems where $\theta_\gamma(u)$ and $f(u,t)$ are convex (e.g., for Bilevel Convex Programs, see Proposition 8), valid bounds can also be computed by exploiting convexities. For example, in this case the function $\varphi_\gamma(z)$ is concave and if $\text{red}_\gamma M = [p', q']$ then we can assume $\varphi_\gamma(q') > 0$ (for $\varphi_\gamma(q') = 0$ would imply that the minimum of $f(z)$ over M is just $f(q')$). From the definition of $\alpha_1, \ldots, \alpha_m$, (see Proposition 11) and q'_s (see (24) and (23)), we have

$$\varphi_\gamma(q' - \alpha_i(q'_i - p_i)e^i) \geq 0 \ (i = 1, \ldots, m), \qquad \varphi_\gamma(q' - \theta_\gamma(\hat{q}')e^s) \geq 0.$$

Therefore the set of all $z \in [p', q']$ satisfying $\varphi_\gamma(z) = 0$ (i.e., the feasible portion in $[p, q]$) is contained in the set

$$S = \left\{ z \in [p', q'] \mid \frac{q'_s - z_s}{q'_s - \theta_\gamma(\hat{q}')} + \sum_{i=1}^{m} (q'_i - z_i)/\alpha_i(q'_i - p_i) \geq 1 \right\}.$$

That is, a valid lower bound for $f(z)$ over the feasible portion in M, is the value $\min\{f(z) \mid z \in S\}$. But, since $f(z)$ is decreasing, its minimum over S is attained at a point z satisfying

$$\frac{q'_s - z_s}{q'_s - \theta_\gamma(\hat{q}')} + \sum_{i=1}^{m} (q'_i - z_i)/\alpha_i(q'_i - p_i) = 1,$$

i.e., such that

$$z_s = \theta_\gamma(\hat{q}') + (q'_s - \theta_\gamma(\hat{q}')) \left[\sum_{i=1}^{m} (q'_i - z_i)/\alpha_i(q_i - p_i) \right].$$

Thus, a lower bound for $f(z)$ over the feasible portion in $[p, q]$ is the value

$$\beta(M) = \min F(x, y) \quad \text{s.t.}$$
$$\left| \begin{array}{l} g_1(x, y) \leq 0, \quad g_2(C(x), y) \leq 0, \\ u \geq C(x), \quad t \geq d(y), \ \ x \geq 0, \ \ y \geq 0, \\ p' \leq (u, t) \leq q', \\ t = \theta_\gamma(\hat{q}') + (q'_s - \theta_\gamma(\hat{q}')) \left[\sum_{i=1}^{m} (q'_i - u_i)/\alpha_i(q'_i - p_i) \right]. \end{array} \right. \tag{28}$$

Note that $\beta(M) \geq f(q')$, since $f(q')$ is the optimal value of the following relaxation of (28):

$$\min F(x, y) \quad \text{s.t.}$$
$$\left| \begin{array}{ll} g_1(x, y) \leq 0, & g_2(C(x), y) \leq 0, \\ \hat{q}' \geq C(x), & q'_s \geq d(y), \ \ x \geq 0, \ \ y \geq 0. \end{array} \right.$$

Thus, for Bilevel Convex Programming a valid lower bound $\beta(M)$ can be computed by solving a convex program (a linear program in the case of (BLP)).

Remark 5 As we saw by Proposition 11, given any box $M := [p,q] \subset [a,b]$, and the current best value γ, if $\varphi_\gamma(q) < 0$ (in particular, if $\theta(\hat{q}) = +\infty$, i.e., $\hat{q} \notin U$), or if $f(q) > \gamma$ (in particular, if $f(q) = +\infty$, i.e., $q \notin W$) then this box should be deleted.

During the process of the Basic BRB Algorithm, a box $M = [p,q]$ is considered only if $\varphi_\gamma(q) \geq 0$ and $f(q) \leq \gamma$. Let $[p',q'] = \text{red}_\gamma[p,q]$ be determined according to (14) and (26), (27). For Bilevel Convex Programs, all the subproblems involved are convex (or linear in the case of (BLP)). A valid lower bound $\beta(M)$ can then be computed by solving the subproblem (28), and the algorithm proceeds further as described in Sect. 5.4.

7 Discussion

7.1 Cases when the dimension can be reduced

Suppose that $m > n_1$ while the mapping C satisfies

$$x' \geq x \Rightarrow C(x') \geq C(x). \tag{29}$$

In that case, the dimension of problem (Q) can be reduced. For this, define

$$\theta(v) = \min\{d(y)|\ g_2(C(v),y) \leq 0,\ y \in \mathbb{R}^{n_2}_+\}, \tag{30}$$

$$\begin{aligned} f(v,t) = \min\{&F(x,y)|\ g_1(x,y) \leq 0,\ g_2(C(v),y) \leq 0, \\ &v \geq x,\ t \geq d(y),\ x \in \mathbb{R}^{n_1}_+,\ y \in \mathbb{R}^{n_2}_+\}. \end{aligned} \tag{31}$$

It is easily seen that these functions are decreasing. Indeed, $v' \geq v$ implies $C(v') \geq C(v)$ and hence $g_2(C(v'),y) \leq 0$, i.e., whenever v is feasible to (30) and $v' \geq v$ then v is also feasible. This shows that $\theta(v') \leq \theta(v)$. Similarly, whenever (v,t) is feasible to (31) and $(v',t') \geq (v,t)$ then (v,t) is also feasible, and hence $f(v',t') \leq f(v,t)$.

With $\theta(v)$ and $f(v,t)$ defined that way, Proposition 9 remains true:

Proposition 16 *Problem* (GBP) *is equivalent to*

$$\min\{f(v,t)|\ t - \theta(v) \leq 0\} \tag{$\tilde{Q}$}$$

and if $(\bar{v},\bar{t})$ *solves* (Q) *then any optimal solution* $(\bar{x},\bar{y})$ *of the problem*

$$\min\{F(x,y)|\ g_1(x,y) \leq 0, g_2(C(x),y) \leq 0, \bar{v} \geq x, \bar{t} \geq d(y), x \geq 0, y \geq 0\}$$

solves (GBP).

Proof If (x,y) is feasible to (GBP) then $g_1(x,y) \leq 0,\ g_2(C(x),y) \leq 0$ and taking $v = x,\ t = d(y) = \theta(x)$ we have $t - \theta(v) = 0$, hence, setting $G := \{(v,t)|\ t - \theta(v) \leq 0\}$ yields

$$\begin{aligned} F(x,y) \geq \min\{&F(x',y')|\ g_1(x',y') \leq 0,\ g_2(C(x'),y') \leq 0, \\ &v \geq x',\ t \geq d(y'),\ x' \geq 0, y' \geq 0\} \\ = f&(v,t) \geq \min\{f(v',t')|\ (v',t') \in G\} = \min(\tilde{\text{Q}}). \end{aligned}$$

Conversely, if $t-\theta(v)\le 0$, then the inequalities $v\ge x,\ t\ge d(y)$ imply $d(y)\le t\le \theta(v)\le\theta(x)$, hence

$$\begin{aligned}
f(v,t) &= \min\{F(x,y)|\ g_1(x,y)\le 0,\ g_2(C(v),y)\le 0,\\
&\qquad v\ge x,\ t\ge d(y),\ x\ge 0,\ y\ge 0\}\\
&\ge \min\{F(x,y)|\ g_1(x,y)\le 0,\ g_2(C(x),y)\le 0,\\
&\qquad \theta(x)\ge d(y),\ \ x\ge 0,\ y\ge 0\}\\
&= \min(\mathrm{GBP}).
\end{aligned}$$

Consequently, min(GBP)= min ($\tilde{\mathrm{Q}}$). The last assertion of the proposition follows. □

Thus, under assumption (29), (GBP) can be solved by the same method as previously, with $v\in\mathbb{R}^{n_1}$ as main parameter instead of $u\in\mathbb{R}^m$. Since $n_1<m$, the method will work in a space of smaller dimension.

More generally, if C is a linear mapping with rank $C=r<m$ such that $E=\mathrm{Ker}C$ satisfies:

$$Ex'\ge Ex\Rightarrow Cx'\ge Cx, \tag{32}$$

then one can arrange the method to work basically in a space of dimension r rather than m. In fact, by writing $E=[E_B,E_N],\ x=\begin{bmatrix}x_B\\ x_N\end{bmatrix}$, where E_B is an $r\times r$ nonsingular matrix, we have, for every $v=Ex$:

$$x=\begin{bmatrix}E_B^{-1}\\ 0\end{bmatrix}v+\begin{bmatrix}-E_B^{-1}E_Nx_N\\ x_N\end{bmatrix}.$$

Hence, setting $Z=\begin{bmatrix}E_B^{-1}\\ 0\end{bmatrix}$, $z=\begin{bmatrix}-E_B^{-1}E_Nx_N\\ x_N\end{bmatrix}$ yields

$$x=Zv+z\quad\text{with}\quad Ez=-E_Nx_N+E_Nx_N=0.$$

Since, in view of (32), $Ez=0$ implies $Cz=0$, it follows that $Cx=C(Zv)$. Now, define

$$\theta(v)=\min\{d(y)|\ g_2(C(Zv),y)\le 0,\ y\in\mathbb{R}^{n_2}_+\}, \tag{33}$$

$$\begin{aligned}
f(v,t)=\min\{F(x,y)|\ g_1(x,y)\le 0,\ g_2(C(Zv),y)\le 0,&\\
v\ge Ex,\ t\ge d(y),\ x\in\mathbb{R}^{n_1}_+,\ y\in\mathbb{R}^{n_2}_+\}.&
\end{aligned}\tag{34}$$

For any $v'\in\mathbb{R}^r=E(\mathbb{R}^{n_1})$, we have $v'=Ex'$ for some x', so $v'\ge v$ implies that $Ex'\ge Ex$, hence, by (32), $Cx'\ge Cx$, i.e., $C(Zv')\ge C(Zv)$, and therefore, $g_2(C(Zv'),y)\le g_2(C(Zv),y)\le 0$. It is then easily seen that $\theta(v')\le\theta(v)$ and $f(v',t')\ge f(v,t)$ for $v'\ge v$ and $t'\ge t$, i.e., both $\theta(v)$ and $f(v,t)$ are decreasing. Furthermore, if $(\bar v,\bar t)$ solves the problem

$$\min\{f(v,t)|\ t-\theta(v)\le 0\} \tag{35}$$

(so that, in particular, $\bar t=\theta(\bar v)$), and $(\bar x,\bar y)$ solves the problem (34) where $v=\bar v,\ t=\bar t=\theta(\bar v)$, then, since $\bar v\ge E\bar x$, we have $\theta(E\bar x)\ge\theta(\bar v)\ge d(\bar y)$. Noting that $Z\bar v=ZE\bar x=\bar x$, this implies that $\bar y$ solves $\min\{d(y)|\ g_2(C\bar x,y)\le 0\}$, and consequently, $(\bar x,\bar y)$ solves (GBP).

Thus, under condition (32), (GBP) can be solved by solving (35) with $\theta(v), f(v,t)$ defined as in (33) and (34).

7.2 Relation to multiobjective programming

The close relationship between BLP and Multiobjective Linear Programming has been noticed very early. However, many earliest results on this subject (1) were later shown to be incorrect (6).

The equivalence between (BLP) and the problem of optimization over the efficient set (OES) was established in (17) (see also (19)). We now prove the equivalence between (GBP) and the general problem (OES).

First, given a mapping $C\colon \mathbb{R}^n \to \mathbb{R}^m$, any problem

$$\min\{f(x) |\, x \in X_E\},$$

where X is a compact set in $\mathbb{R}^n$, and X_E denotes the efficient set of X relative to V-min$C(x)$ can be rewritten as the following special (GBP):

$$\min f(y) \quad \text{s.t. } y \in X \text{ solves } \min\left\{\sum_{i=1}^{m} C_i(x) |\, C(x) \leq C(y),\ x \in X\right\}.$$

Conversely, for any given (GBP), it is easy to see that if we define

$$\widetilde{C}(x) = \left(C(x), -\sum_{i=1}^{m} C_i(x), d(y)\right),$$

then (GBP) reduces to the following problem of constrained optimization over the efficient set (37):

$$\min\{F(x,y) |\, g_1(x,y) \leq 0,\ (x,y) \in X_E\}, \tag{36}$$

where $X = \{(x,y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+ |\, g_2(C(x),y) \leq 0\}$ and X_E denotes the efficient set of X relative to V-min$\widetilde{C}$. Indeed, $(\bar{x},\bar{y}) \in X$ is feasible to (GBP) if and only if $g_1(\bar{x},\bar{y}) \leq 0$, and for any $(x,y) \in X$ the relations

$$C(x) \leq C(\bar{x}),\ -\sum_{i=1}^{m} C_i(x) \leq -\sum_{i=1}^{m} C_i(\bar{x}), \quad d(y) \leq d(\bar{y}),$$

imply $C(x) = C(\bar{x}),\ d(y) = d(\bar{y})$, which shows that (x,y) is an efficient point of X relative to V-min$\widetilde{C}$ and satisfying $g_1(x,y) \leq 0$.

A peculiar feature worth noting of the above problem (OES) is that any weakly efficient point is efficient, while the functions $\widetilde{C}_i(x), i = 1,\ldots,m+2$, are not linearly independent. Therefore, it would be of interest to see how to solve (GBP) using the Basic BRB Algorithm proposed in (37) for constrained optimization over the weakly efficient set.

7.3 Specialization to (BLP)

As we saw in Sect. 3, the BLP problem

$$\begin{array}{l} \min L(x,y) \quad \text{s.t.} \\ A_1x + B_1y \geq c^1, \quad x \in \mathbb{R}^{n_1}_+,\ \ y \text{ solves} \\ \min\{\langle d,y\rangle |\, A_2x + B_2y \geq c^2, \quad y \in \mathbb{R}^{n_2}_+\}, \end{array} \tag{BLP}$$

fulfills the general assumptions formulated for (GBP), provided the set $\{(x,y) \in \mathbb{R}^{n_1}_+ \times \mathbb{R}^{n_2}_+ |\, A_1x + B_1y \geq c^1,\ A_2x + B_2y \geq c^2\}$ is a nonempty polytope, By specializing the basic operations to the case $F(x,y) = L(x,y)$, $d(y) = \langle d,y\rangle$, $C(x) = A_2x$, while $g_1(x,y) = \max_{i=1,\dots,m_1}(c^1 - A_1x - B_1y)_i$, $g_2(u,y) = \max_{i=1,\dots,m_2}(c^2 - u - B_2y)_i$, with $c^i \in \mathbb{R}^{m_i}$, $i = 1,2$, the Basic BRB Algorithm formulated in Sect. 5.4 can be used to solve (BLP).

Define

$$A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \qquad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \qquad c = \begin{bmatrix} c^1 \\ c^2 \end{bmatrix},$$
$$A_1 = \begin{bmatrix} A_{1,1} \\ \dots \\ A_{1,m_1} \end{bmatrix}, \qquad A_2 = \begin{bmatrix} A_{2,1} \\ \dots \\ A_{2,m_1} \end{bmatrix},$$
$$D = \{(x,y)|\ Ax + By \geq c,\ x \geq 0,\ y \geq 0\},$$

$$\theta(u) = \min\{\langle d,y\rangle|\ (x,y) \in D,\ u \geq A_2x\}, \quad z = (u,t),$$
$$f(z) = \min\{L(x,y)|\ (x,y) \in D,\ u \geq A_2x,\ t \geq \langle d,y\rangle\}.$$

- *Initial box*
  The initial box $[a,b] \subset \mathbb{R}^{m_2+1}$ is defined by formula

$$a_i = \min\{\langle A_{2,i}, x\rangle|\ (x,y) \in D\}, \quad i = 1,\dots,m_2,$$
$$a_{m_2+1} = \min\{\langle d,y\rangle|\ (x,y) \in D\}$$
$$b_i = \max\{\langle A_{2,i}, x\rangle|\ (x,y) \in D\}, \quad i = 1,\dots,m_2,$$
$$b_{m_2+1} = \max\{\langle d,y\rangle|\ (x,y) \in D\}.$$

- *Reduction operation*

Let $\bar{z} =$ CBS and $\gamma = f(\bar{z}) =$ CBV. If $M := [p,q]$ is any subbox of $[a,b]$ still of interest at a given iteration, then a valid reduction of M, written $\text{red}_\gamma M$, is determined as follows, where $\varphi_\gamma(z) = t - \theta_\gamma(u)$ for $z = (u,t)$:

(1) If $\varphi_\gamma(q) < 0$, or $f(q) > \gamma$ then $\text{red}_\gamma M = \emptyset$ (M is to be deleted);
(2) Otherwise, $\text{red}_\gamma M = [p', q']$ is determined by formulas (24), (23), where α_i, $\bar{\beta}_i$, $i = 1,\dots,m$, are computed as follows:

Computing $\bar{\beta}_i$, $i = 1,\dots,m_2$ (see Proposition 15)

$$\bar{\beta}_i = \min_{x,y,\beta} \beta$$
$$\text{s.t.} \left| \begin{array}{l} A_2x + B_2y \geq c^2,\ (p_1,\dots,p_{m_2}) + \beta(q_i - p_i)e^i \geq A_2x, \\ p_{m_2+1} \geq \langle d,y\rangle + \varepsilon, \\ x \geq 0, \quad y \geq 0, \quad 0 \leq \beta \leq 1, \end{array} \right.$$

where $\varepsilon > 0$ is a small number, e.g., $\varepsilon = 0.01(q_i - p_i)$.

Computing α_i, $i = 1,\dots,m_2$:

$$\alpha_i = \max_{x,y,\alpha} \alpha$$
$$\text{s.t.} \left| \begin{array}{l} L(x,y) \leq \gamma \\ Ax + By \geq c \\ (q'_1,\dots,q'_{m_2}) - \alpha(q'_i - p_i)e^i \geq A_2x, \\ q'_{m_2+1} \geq \langle d,y\rangle, \\ x \geq 0, \quad y \geq 0, \ 0 \leq \alpha \leq 1. \end{array} \right.$$

- *Lower bounding*

For any box $M = [p, q]$ with $\text{red}_M = [p', q']$ a lower bound for $f(z)$ over $[p, q]$ is

$\beta(M) = \min\{f(z) |\ z \in S\}$

$$= \begin{array}{l} \min L(x, y) \\ \text{s.t.} \left| \begin{array}{l} Ax + By) \geq c, \\ u \leq A_2 x, \qquad t \geq \langle d, y \rangle, \quad x \geq 0, \ \ y \geq 0, \\ p' \leq (u, t) \leq q'. \\ t = \theta_\gamma(\hat{q}') + \left(q'_{m_2+1} - \theta_\gamma(\hat{q}')\right) \left[\sum_{i=1}^{m_2} (q'_i - u_i)/\alpha_i(q'_i - p_i)\right], \end{array} \right. \end{array}$$

where α_i are defined as above.

8 Illustrative examples

To illustrate how the algorithm works we present some numerical examples. The algorithm was coded in C++ and run on a PC Pentium IV 2.53 GHz, RAM 256 Mb DDR, using CPLEX 8.0 for solving linear subproblems. Some details of computation are shown for the three first examples. The last three examples 5–7, taken from (16), are very simple problems that can be solved in next to no time by the present algorithm, confirming or correcting an optimal solution indicated in (16).

Example 1 With tolerance 0.0001 solve the problem

$$\begin{array}{l} \min(x^2 + y^2) \quad \text{s.t.} \\ x \geq 0, \ y \geq 0, \quad y \text{ solves} \\ \min(-y) \quad \text{s.t.} \\ 3x + y \leq 15, \\ x + y \leq 7, \\ x + 3y \leq 15. \end{array}$$

This problem can be viewed as a (GBP) satisfying conditions (A1)–(A3), with $n_1 = n_2 = 1,\ m = 1$, and

$$F(x, y) = x^2 + y^2, \qquad g_1(x, y) = \max\{x + y - 7,\ x + 3y - 15\},$$
$$d = -1, \quad g_2(C(x), y) = -C(x) + y - 15, \quad C(x) = -3x.$$

Here $D = \{(x, y) \in \mathbb{R}^2_+ |\ 3x + y \leq 15,\ x + y \leq 7,\ x + 3y \leq 15\}$,

$$\begin{array}{l} \theta(u) = \min(-y) \quad \text{s.t.} \\ (x, y) \in D, \\ -3x \leq u, \end{array}$$

$$\begin{array}{l} f(u, t) = \min(x^2 + y^2) \quad \text{s.t.} \\ (x, y) \in D, \\ -3x \leq u, \quad -y \leq t. \end{array}$$

Initialization

$$a_1 = \min\{-3x|\ (x,y) \in D\} = -15, \qquad b_2 = \theta(-15) = 0,$$
$$b_1 = \max\{-3x|\ (x,y) \in D\} = 0, \qquad a_2 = \theta(0) = -5.$$

Initial box $M_1 = [a,b]$, with $a = (-15,-5)$, $b = (0,0)$. Lower bound over M_1 : $\beta(M_1) = 12.5$; CBV $= 25$.

Iteration 1
M_1 is divided into two boxes M_{11}, M_{12}.
$M_{11} = [(-7.5,-5),\ (0,0)]$, with
redM$_{11} = [(-7.5,-5),\ (0,-4.166667)]$; $\beta(M_{11}) = 22.5$.
$M_{12} = [(-15,-5),\ (-7.5,0)]$, with
redM$_{12} = [(-15,-4.166667),\ (-7.5,0)]$; $\beta(M_{12}) = 18.382353$.
CBS $= (2.5, 4.166667)$ with CBV $= 23.611111$.
M_{12} is selected (for further subdivision): reset $M_2 \leftarrow M_{12}$.

Iteration 2
Number of active nodes: 2.
CBV $= 23.611111$, $\min_M\{\beta(M)\} = 18.382353$
M_2 is divided into two boxes M_{21}, M_{22}.
$M_{21} = [(-11.25,-4.166667),\ (-7.5,0)]$, with
redM$_{21} = [(-11.25,-4.166667),\ (-7.5,-3.25)]$; $\beta(M_{21}) = 23.410405$.
$M_{22} = [-(-15,-4.166667),\ (-11.25,0)]$, with
redM$_{22} = [(-15,-3.25),\ (-11.25,0)]$; $\beta(M_{22}) = 21.778351$.
M_{22} is selected: reset $M_3 \leftarrow M_{22}$.

Iteration 3
Number of active nodes: 3.
CBV $= 23.611111$, $\min_M\{\beta(M)\} = 21.778351$
M_3 is divided into two boxes: M_{31}, M_{32}.
$M_{31} = [(-13.125,-3.25),\ (-11.25,0)]$, with
redM$_{31} = [(-13.125,-3.25),\ (-11.25,-1.875)]$; $\beta(M_{31}) = 22.645548$.
$M_{32} = [-(-15,-3.25),\ (-13.125,0)]$, with
redM$_{32} = [(-15,-1.875),\ (-13.125,0)]$, $\beta(M_{12}) = 22.5$.
CBS $= (4.375, 1.875)$ with CBV $= 22.65625$.
M_{32} is selected: reset $M_4 \leftarrow M_{32}$.
.
.
.
Iteration 10
Number of active nodes: 1.
CBV $= 22.504340$, $\min_M\{\beta(M)\} = 22.5$.
M_{10} is divided into two boxes: M_{101}, M_{102}.
$M_{101} = [(-13.476563,-1.640625),\ (-13.359375,-1.406250)]$, with
redM$_{101} = [(-13.476563,-1.640625),\ (-13.359375,-1.523438)]$;
$\beta(M_{101}) = 22.500610$.
$M_{102} = [(-13.593750,-1.640625),\ (-13.476563,-1.406250)]$, with
redM$_{102} = [(-13.593750,-1.523438),\ (-13.476563,-1.406250)]$;
$\beta(M_{102}) = 22.5$.
CBS $= (4.492188, 1.523438)$ with CBV $= 22.500610$.

With tolerance $\varepsilon = 0.0001$, these two boxes are deleted. Since no box remains for exploration, the algorithm terminates.

Computational results:
Optimal solution: (4.492188, 1.523438)
Optimal value: 22.500610 (relative error $\leq$ 0.0001)
Computational time: 0.015 s.
Optimal solution found at iteration 10 and confirmed at iteration 10
Maximal number of nodes: 4

Example 2 (Test problem 7 in (16), Chap. 9)

$$\begin{aligned}
&\min(-8x_1 - 4x_2 + 4y_1 - 40y_2 + 4y_3) \quad \text{s.t.}\\
&x_1, x_2 \geq 0, \quad y \text{ solves}\\
&\min(y_1 + y_2 + 2y_3) \quad \text{s.t.}\\
&-y_1 + y_2 + y_3 \leq 1,\\
&2x_1 - y_1 + 2y_2 - 0.5y_3 \leq 1,\\
&2x_2 + 2y_1 - y_2 - 0.5y_3 \leq 1,\\
&y_1, y_2, y_3 \geq 0.
\end{aligned}$$

Here $g_1(x,y) = -y_1 + y_2 + y_3 - 1,\ C(x) = -(x_1, x_2)^T$,
$g_2(u,y) = \max\{-2u_1 - y_1 + 2y_2 - 0.5y_3 - 1,\ -2u_2 + 2y_1 - y_2 - 0.5y_3 - 1\}$, so

$$\begin{aligned}
\theta(u) = \min(y_1 + y_2 + 2y_3) \quad &\text{s.t.}\\
&-y_1 + y_2 + y_3 \leq 1,\\
&2x_1 - y_1 + 2y_2 - 0.5y_3 \leq 1,\\
&2x_2 + 2y_1 - y_2 - 0.5y_3 \leq 1,\\
&-x_1 \leq u_1,\ -x_2 \leq u_2\\
&x_1, x_2, y_1, y_2, y_3 \geq 0.
\end{aligned}$$

$$\begin{aligned}
f(u,t) = \min(-8x_1 - 4x_2 + 4y_1 - 40y_2 + 4y_3) \quad &\text{s.t.}\\
&-y_1 + y_2 + y_3 \leq 1,\\
&2x_1 - y_1 + 2y_2 - 0.5y_3 \leq 1,\\
&2x_2 + 2y_1 - y_2 - 0.5y_3 \leq 1,\\
&-x_1 \leq u_1,\ -x_2 \leq u_2,\ y_1 + y_2 + 2y_3 \leq t\\
&x_1, x_2, y_1, y_2, y_3 \geq 0.
\end{aligned}$$

Initialization
Initial box $M_1 = [(-3, -1.8, 0), (0, 0, 5)]$ with red$M_1 = [(-2.64, -1.8, 0),\ (0, 0, 5)]$
CBS $= (0.5, 0.5, 0, 0, 0)$, CBV $= -6$, $\beta(M) = -50$.
Iteration 1
M_1 is divided into two boxes:
$M_{11} = [(-1.32, -1.8, 0), (0, 0, 5)]$ with $\beta(M_{11}) = -50$.
$M_{12} = [(-2.64, 1.8, 0), (-1.32, 0, 5)]$ with red$M_{12} = [(-2.64, -1.536, 0.32), (-1.32, 0, 5)]$ and $\beta(M_{12}) = -28$.
M_{11} is selected (for further subdivision). reset $M_2 \leftarrow M_{11}$.

Iteration 2
Number of active nodes: 2.
CBV $= -6$, $\min_M\{\beta(M)\} = -50$.

M_2 is divided into two boxes:
$M_{21} = (-1.32, -0.9, 0), (0, 0, 5)]$ with
redM$_{21} = [(-1.32, -0.9, 0), (0, 0, 0.86)]$ and $\beta(M_{21}) = -25.64$.
$M_{22} = (-1.32, -1.8, 0), (0, -0.9, 5)]$ with
redM$_{22} = (-1.32, -1.8, 0), (0, -0.9, 3.2)]$ and $\beta(M_{22}) = -38$.
M_{22} is selected: reset $M_3 \leftarrow M_{22}$.

Iteration 3
Number of active nodes: 3.
CBV $= -6$, $\min_M\{\beta(M)\} = -38$.
M_3 is divided into two boxes:
$M_{31} = [(-0.66, -1.8, 0), (0, -0.9, 3.2)]$ with $\beta(M_{31}) = -38$.
$M_{32} = [(-1.32, -1.8, 0), (-0.66, -0.9, 3.2)]$ with
redM$_{32} = (-1.32, -1.668, 0), (-0.66, -0.9, 3.2)]$ and $\beta(M_{32}) = -27$.
M_{31} is selected: reset $M_4 \leftarrow M_{31}$.

Iteration 4
Number of active nodes: 4.
CBV $= -6$, $\min_M\{\beta(M)\} = -38$.
M_4 is divided into two boxes:
$M_{41} = [(-0.66, -1.35, 0), \ (0, -0.9, 3.2)]$ with
redM$_{41} = [(-0.66, -1.35, 0), (0, -0.9, 0.71)]$ and $\beta(M_{41}) = -24.82$.
$M_{42} = [(-0.66, -1.8, 0), \ (0, -1.35, 3.2)]$ with
redM$_{41} = (-0.66, -1.8, 0.35), (0, -1.35, 2.3)]$ and $\beta(M_{42}) = -32$.
CBS is updated:
CBS $= (0.15, 0.675, 0, 0.35, 0)$ with CBV $= -17.9$.
.
.
.
Iteration 22
Number of active nodes: 1.
CBV $= -25.859375$, $\min_M\{\beta(M)\} = -26.187500$.
M_{22} is divided into two boxes:
$M_{221} = [(-0.009844, -1.792969, 1.357813), (0, -1.785938, 1.428125)]$ with
redM$_{221} = [(-0.009844, -1.792969, 1.357813), (0, -1.785938, 1.388750)]$
$\beta(M_{221}) = -26.003750$.
$M_{222} = [(-0.009844, -1.8, 1.357813), (0, -1.792969, 1.428125)]$ with
redM$_{222} = [(-0.009844, -1.8, 1.378906), (0, -1.792969, 1.414063)]$
$\beta(M_{222}) = -26.093750$.
CBS is updated:
CBS $= (0, 0.896484, 0, 0.597656, 0.390625)$ with CBV $= -25.929688$.
With tolerance $\varepsilon = 0.01$, these two boxes are deleted. Since no box remains for exploration, the algorithm terminates.

Computational results:
Optimal solution: $x = (0, 0.896484), y = (0, 0.597656, 0.390625)$.
Optimal value: -25.929688 (relative error ≤ 0.01)
Computational time: 0.047 s.
Optimal solution found at iteration 22 and confirmed at iteration 22.
Maximal number of nodes: 9

Example 3 This example is a (BLP) randomly generated with $n_1 = 10$ outer variables, $n_2 = 6$ inner variables, $m_1 = 2$ outer constraints, and $m_2 = 7$ inner constraints.

$$L(x, y) = 12x_1 - x_2 - 12x_3 + 13x_4 + 2x_6 - 5x_8 + 6x_9 - 11x_{10} - 5y_1 - 6y_2 - 4y_3 - 7y_4$$

$$\begin{aligned}
A_1 &= \begin{bmatrix} 2 & 3 & -14 & 2 & 9 & -2 & -1 & 4 & 0 & -2 \\ -1 & 7 & -13 & 0 & 15 & -2 & 8 & 4 & -4 & 7 \end{bmatrix}, \\
B_1 &= \begin{bmatrix} 3 & -9 & 2 & 8 & -1 & 8 \\ 6 & 2 & -6 & -2 & -8 & 4 \end{bmatrix}, \\
c^1 &= (-30, 134)^T, \\
d &= (3, -2, -3, -3, 1, 6)^T, \\
A_2 &= \begin{bmatrix} 5 & -7 & -4 & 2 & -3 & 9 & -9 & 1 & 3 & -11 \\ -6 & 5 & 3 & 2 & -8 & -5 & -8 & 3 & -7 & -3 \\ 6 & 4 & -2 & 0 & 2 & -3 & 3 & -2 & -2 & -4 \\ -5 & -6 & 0 & 4 & -3 & 8 & -1 & 0 & -2 & 3 \\ -11 & 11 & -4 & -5 & 10 & 6 & -14 & 7 & 11 & 3 \\ -9 & 12 & 4 & 10 & -2 & -8 & -5 & 11 & 4 & -1 \\ -7 & 2 & 6 & 0 & 11 & -1 & 2 & 2 & 1 & 2 \end{bmatrix}, \\
B_2 &= \begin{bmatrix} -10 & 9 & 6 & -4 & -6 & 3 \\ 5 & 7 & -1 & -1 & 6 & -4 \\ -10 & -5 & -6 & 4 & -3 & 1 \\ 4 & 3 & 4 & 4 & -1 & -1 \\ 10 & 7 & -7 & -7 & -2 & -7 \\ -2 & 5 & -10 & -1 & -4 & -5 \\ 5 & 5 & 6 & 5 & -1 & 12 \end{bmatrix}, \\
c^2 &= (-83, -92, -168, 96, 133, -89, 192)^T.
\end{aligned}$$

Computational results
Initial box $M_1 = [0, 10]^{16}$
Optimal solution:
$x = (0, 8.170692, 10, 0, 7.278940, 3.042311, 0, 10, 0.001982, 9.989153)^T$.
$y = (3.101280, 10, 10, 10, 0, 9.846133)^T$
Optimal value: -447.461263 (relative error ≤ 0.01)
Computational time: 54.907 s.
Optimal solution found at iteration 9033 and confirmed at iteration 10323
Maximal number of nodes: 1442

Example 4 (Test problem 3 in [14], Chap. 9.3)

$$\begin{aligned}
&\min(2x_1 + 2x_2 - 3y_1 - 3y_2 - 60) \text{ s.t.} \\
&0 \le x_1, x_2 \le 50,\ -10 \le y_1, y_2 \le 20,\ y \text{ solves} \\
&\min_y (y_1 - x_1 + 20)^2 + (y_2 - x_2 + 20)^2 \text{ s.t.} \\
&x_1 + x_2 + y_1 - 2y_2 - 40 \le 0, \\
&-x_1 + 2y_1 \le -10, \\
&-x_2 + 2y_2 \le -10.
\end{aligned}$$

Computational results:
Optimal solution: $x = (0, 0), y = (-10, -10)$.
Optimal value: 0 (relative error ≤ 0.01).

Computational time: 0.266 s.
Optimal solution found at iteration 1 and confirmed at iteration 43.
Maximal number of nodes: 3
The above optimal solution was given in (16) as a best known solution only. A local solution $x = (25, 30), y = (5, 10)$ was reported in (26) as the solution of the problem.

Example 5 (Test problem 4, Chap. 9.2, in (16)) This problem is taken from (12) and has been used for testing purposes in (42). The best known solution given in (16) is confirmed as optimal by the present algorithm.

$$\begin{aligned}
&\min(x - 4y) \quad \text{s.t.}\\
&x \geq 0, \quad y \text{ solves}\\
&\min(y) \quad \text{s.t.}\\
&\quad 2x - y \geq 0,\\
&-2x - 5y \geq -108,\\
&-2x + 3y \geq 4,\\
&y \geq 0.
\end{aligned}$$

Computational results
Optimal solution: $x = 18.929688, y = 13.953125$.
Optimal value: -36.882813 (relative error $\leq$0.01)
Computational time: 0 s.
Optimal solution found at iteration 8 and confirmed at iteration 8
Maximal number of nodes: 2

Example 6 (Test problem 8 in (16), Chap. 9.2). This problem is taken originally from (4), where, however, a nonoptimal solution was claimed to be optimal. A correct optimal solution, pointed out later in (29), is confirmed by the present algorithm.

$$\begin{aligned}
&\min(-2x_1 + x_2 + 0.5y_1) \quad \text{s.t.}\\
&x_1 + x_2 \leq 2, \ x \geq 0, \quad y \text{ solves}\\
&\min(-4y_1 + y_2) \quad \text{s.t.}\\
&\quad 2x_1 - y_1 + y_2 \geq 2.5,\\
&-x_1 + 3x_2 - y_2 \geq -2,\\
&y \geq 0.
\end{aligned}$$

Computational results
Optimal solution: $x = (2, 0), y = (1.5, 0)$.
Optimal value: -3.25 (relative error ≤ 0.01)
Computational time: 0.093 s.
Optimal solution found at iteration 74 and confirmed at iteration 107
Maximal number of nodes: 37

Example 7 (Test problem 5 in (16), Chap. 9.2)

$$\begin{aligned}
&\min(-x + 10y_1 - y_2) \text{ s.t.}\\
&x \in R_+, \ y \in R^2_+, \ y \text{ solves}\\
&\min_y(-y_1 - y_2) \text{ s.t.}\\
&x - y_1 \leq 1,\\
&x + y_2 \leq 1,\\
&y_1 + y_2 \leq 1.
\end{aligned}$$

Computational results
Optimal solution: $x = 0, y = (0, 1)$.
Optimal value: -1
Computational time: 0 s.
Optimal solution found at iteration 0 and confirmed at iteration 0.
Maximal number of nodes: 1
Note that the best known solution $x = 1, y = (0, 0)$ given in (16) is not even feasible. It is actually an optimal solution of the problem with the inner objective function changed to $y_1 + y_2$.

9 Conclusion

In this paper a novel approach to bilevel nonlinear programming is developed that is based on exploiting the monotonic structure underlying this class of problems. The advantage of this approach is that it permits efficient reduction and bounding in the framework of a BRB procedure using monotonicity cuts. Once more the versatility of monotonic optimization is demonstrated as a unified method for approaching a wide class of difficult nonconvex global optimization problems.

References

1. Bard, J.F.: An efficient point algorithm for a linear two-stage optimization problem. Oper. Res. **31**, 670–684 (1983)
2. Bard, J.F.: Convex two-level optimization. Math. Program. **40**, 15–27 (1988)
3. Bard, J.F.: Practical bilevel optimization algorithms and Optimization. Kluwer, Dordrecht (1998)
4. Bard, J.F., Falk, J.: An explicit solution to the multi-level programming problem. Comp. Oper. Res. **9**, 77–100 (1982)
5. Baer, E., Hiltner, A., Morgan, R.: Biological and synthetic hierarchical composites. Phys. Today **46**, 60–67 (1992)
6. Ben-Ayed, O.: Bilevel linear programming. Comput. Oper. Res. **20**, 485–501 (1993)
7. Bialas, W.F., Karwan, C.H.: On two-level optimization. IEEE Trans. Auto.Cont. Vol. AC-27, 211–214 (1982)
8. Brackel, J., McGill, J.: Mathematical programs with optimization problems in the constraints. Oper. Res. **21**, 37–44 (1973)
9. Brackel, J., McGill, J.: A method for solving mathematical programs with nonlinear programs in the constraints. Oper. Res. **22**, 1097–1101 (1974)
10. Candler, W., Townsley, R.: A linear two-level programming problem. Comput. Ops Res. **9**, 59–76 (1982)
11. Chen, Y., Florian, M.: Congested O-D trip emand adjustment problem: bilevel programming formulation and optimality conditions. In: Migdalas, A., Pardalos, P., Värbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 1–22 Kluwer, Dordrecht (1988)
12. Clark, P.A., Westerberg, A.W.: A note on the optimality conditions for the bilevel programming problem. Naval Res. Logistics **35**, 413–418 (1988)
13. Dempe, S.: An implicit function approach to bilevel programming problems. In: Migdalas, A., Pardalos, P., Värbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 273–289 Kluwer, Dordrecht (1988)
14. Edmunds, T., Bard, J.: Algorithms for nonlinear bilevel mathematical programming. Trans. Syst. Man Cybernetics **21**, 83–89 (1991)
15. Falk, J.E., Liu, J.: On bilevel programming, Part I; general nonlinear cases. Math. Program. **70**, 47–72 (1995)
16. Floudas, C. et al.: Handbook of Test Problems for Local and Global Optimization. Kluwer, Dordrecht (1999)

17. Fülöp, J.: On the equivalence between a linear bilevel programming problem and linear optimization over the efficient set. Working paper WP 93-1, LORDS, Computer and Automation Institute, Budapest
18. Hansen, P., Jaumard, B., savard, G.: New branch and bound rules for linear bilevel programming. SIAM J. Sci. Sta. Comput. **13**, 1194–1217 (1992)
19. Konno, H., Thach, P.T., Tuy, H.: Optimization on Low Rank Nonconvex structures. Kluwer, Dordrecht (1997)
20. Lignola, M.B., Morgan, J.: Existence of solutions to generalized bilevel programming problem. In: Migdalas, A., Pardalos, P., Värbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 315–330 Kluwer, Dordrecht (1988)
21. Loridan, P., Morgan, J.: Quasiconvex lower level problem and applications to two level optimization. Lecture Notes in Economics and Mathematical Systems, vol. 345, pp. 325–341, Springer, Dordrecht (1990)
22. Migdalas, A., Pardalos, P.M.: Nonlinear bilevel problems with convex second level problem – heuristics and descent methods. In: Du, D.-Z., Zhang, X.-S., Cheng, K., (eds.) Operations Research and Its Applications, pp. 194–204. World Publishing Corporation, New York (1995)
23. Migdalas, A., Pardalos, P.M.: Hierarchical and bilevel programming. J. Glob. Optim. **8**(3) (1996)
24. Migdalas, A., Pardalos, P.M., Värbrand, P. (eds.): Multilevel Optimization: Algorithms and Applications. Kluwer, Dordrecht (1998)
25. Pardalos, P.M., Deng, X.: Complexity issues in hierarchical optimization. In: DIMACs series, vol. 37, American Mathematical Society, pp. 219–224. Providence, RI (1997)
26. Shimizu, K., Ishizuka, Y., Bard, J.F.: Nondifferentiable and Two-Level Mathematical Programming. Kluwer, Dordrecht (1997)
27. Tuy, H.: Bilevel linear programming, multiobjective programming, and monotonic reverse convex programming. In: Migdalas, A., Pardalos, P.M., Värbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 295–314 Kluwer, Dordrecht (1998)
28. Tuy, H.: Convex Analysis and Global Optimization. Kluwer, Dordrecht (1998)
29. Tuy, H., Migdalas, A., Värbrand, P.: A global optimization approach for the linear two-level program. J. Glob. Optim. **3**, 1–23 (1992)
30. Tuy, H., Migdalas, A., Värbrand, P.: A quasiconcave minimization method for solving linear two-level programs. J. Glob. Optim. **4**, 243–263 (1994)
31. Tuy, H., Ghannadan, s.: A new branch and bound method for bilevel linear programs. In: Migdalas, A., Pardalos, P.M., Värbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 231–249 Kluwer, Dordrecht (1998)
32. Tuy, H.: Monotonic optimization: problems and solution approaches. SIAM J. Optim. **11**, 464–494 (2000)
33. Tuy, H.: Convexity and monotonicity in global optimization. In: Hadjisavvas, N., Pardalos, P.M. (eds.) Advances in Convex Analysis and Optimization, pp. 569–594 Kluwer, Dordrecht (2001)
34. Tuy, H.: Hierarchical optimization. In: Pardalos, P., Resende, M. (eds.) Handbook of Applied Optimization, pp. 502–513 Oxford University Press, Oxford (2002)
35. Tuy, H.: Monotonicity in the framework of generalized convexity. In: Eberhard, A., Hadjisavvas, N., Luc, D.T. (eds.) Generalized Convexity, Generalized Monotonicity and Applications, pp. 61–85 Springer, Berlin (2005)
36. Tuy, H., Al-Khayyal, F., Thach, P.T.: Monotonic Optimization: Branch and Cuts Methods. In: Audet, C., Hansen, P., Savard, G. (eds.) Essays and Surveys on Global Optimization, pp. 39–78 Springer, Berlin (2005)
37. Tuy, H., Hoai-Phuong, N.T.: Optimization under composite monotonic constraints and constrained optimization over the efficient set. In: Liberti, L., Maculan, N. (eds.) Global Optimization: from Theory to Implementation, pp. 3–31 Springer, Berlin (2006)
38. Vicente, L.N., Calamai, P.Y.: Bilevel and multilevel programming: a bibliographic review. J. Glob. Optim. **5**, 291–306 (1994)
39. Vicentee, L., Savard, G., Judice, J.: Descent approaches for quadratic bilevel programming. J. Optim. Theory Appl. **81**, 379 (1994)
40. Vincent, J.F.: Structural Biomaterials. Princeton University Press, Princeton (1990)
41. White, D.J., Anandalingam, G.: A penalty function approach for solving bilevel linear programs. J. Glob. Optim. **3**, 397–419 (1993)
42. Yezza, A.: First-order necessary optimality conditions for general bilevel programming problems. J Optim Theory Appl **89**, 189–219 (1996)