Combined Branch-and-Bound and Cutting Plane Methods for Solving a Class of Nonlinear Programming Problems

LÊ D. MUU* and W. OETTLI
Universität Mannheim, D-6800 Mannheim, Germany

(Received: 7 August 1990; accepted: 11 March 1993)

Abstract. We propose unified branch-and-bound and cutting plane algorithms for global minimization of a function $f(x, y)$ over a certain closed set. By formulating the problem in terms of two groups of variables and two groups of constraints we obtain new relaxation bounding and adaptive branching operations. The branching operation takes place in y-space only and uses the iteration points obtained through the bounding operation. The cutting is performed in parallel with the branch-and-bound procedure. The method can be applied implementably for a certain class of nonconvex programming problems.

Key words. Branch-and-bound, cutting plane, decomposition, convex-concave function, global optimization.

1. Introduction

Cutting plane methods are fundamental tools in mathematical programming. The first cutting plane methods were introduced by Cheney and Goldstein [5], and Kelley [17]. Since then several modifications were proposed to improve speed of convergence, to avoid accumulation of constraints, and to handle nondifferentiable functions (see, e.g. [4, 8, 9, 10, 18, 22, 28]). Recently, cutting plane methods have been applied for solving a broad class of nonconvex optimization problems [14, 16, 21, 22, 25, 30, 32].

Another basic approach used widely in integer and nonconvex programming problems are branch-and-bound methods, where sequences of decreasing upper bounds and increasing lower bounds for the optimal value are constructed by successively refined partitions of the feasible region and corresponding estimation procedures.

Falk and Soland [7] developed a branch-and-bound method for solving certain nonconvex programming problems having compact, convex feasible sets and separable objective functions. This method was extended further by Soland in [27] to handle separable nonconvex constraints. Al-Khayyal and Falk [1] used the idea of Falk and Soland [7] to obtain an algorithm for solving biconvex programming problems with jointly convex constraints. McCormick [20] considered the global optimization problem dealing with factorable functions. An underestimating

*On leave from Institute of Mathematics, Hanoi, by a grant from Alexander-von-Humboldt-Stiftung.

Journal of Global Optimization 3: 377–391, 1993.

function of a factorable function was calculated in [20], under which a branch-and-bound algorithm for solving factorable programs was described. A more general branch-and-bound scheme was proposed by Horst [12] which can be used for solving a fairly broad class of nonconvex programming problems. The bounding operations used in these methods were based on underestimating functions of the objective function, most of them used the convex envelope function. In [29] Thoai, Tuy and in [32] Tuy, Thieu, Thai proposed branch-and-bound methods for minimizing a concave function over a polyhedron and over a convex set using a bounding operation based on the fact that the minimum of a concave function on a polyhedron, if finite, is attained at a vertex of the polyhedron. Recently Muu and Oettli [22, 23] have proposed branch-and-bound methods for minimizing indefinite quadratic functions and convex-concave functions over a convex set. The bounding operations used in these methods are based on a suitable relaxation of the constraints. Rather general branch-and-bound schemes can be found in [3, 14, 16, 31].

Numerical experience indicates that in nonconvex optimization problems both cutting plane methods and branch-and-bound methods and their combinations are efficient only for problems with moderate size. To overcome this drawback several decomposition methods were proposed (see, e.g. [2, 23, 25, 26]). Decomposition approaches were used earlier in linear and convex programming by Dantzig and Wolfe [6].

In this paper we shall present a combined cutting plane and branch-and-bound method for solving the following problem:

$$(P) \quad \min\{f(x, y)\colon g(x, y) \leq 0, (x, y) \in S\} ,$$

where f, g: $\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ are continuous, and

$$S := \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^m\colon \varphi(x, y) \leq 0\} ,$$

φ being a convex function on $\mathbb{R}^n \times \mathbb{R}^m$ (hence continuous).

Using the fact that Problem (P) has two groups of variables x and y and two types of constraints $g(x, y) \leq 0$, $(x, y) \in S$ we obtain a new relaxation bounding and adaptive branching operation by using a separation function. By choosing specific forms of this separation function we obtain different bisections; all of them use the iteration points obtained through the bounding operation. The branching operation is performed in y-space only, and therefore it allows us to decompose certain nonconvex problems of the form (P) into convex subprograms in (x, y)-space and nonconvex subprograms in y-space. This suggests to apply the method when the dimension of y-space is relatively small, even though the dimension of x-space may be fairly large.

The cutting, as usually, is introduced to approximate the convex set S by polyhedral convex sets, but here it is performed in parallel with the branch-and-bound procedure. The algorithm first is described as a conceptual method without reference to implementation. We then specialize it to several cases to obtain implementable algorithms for minimizing a convex-concave function over a convex set and solving convex programs with an additional convex-concave constraint.

2. Description of the Algorithm

We denote by G the feasible domain of Problem (P). We assume that we have fixed two compact polyhedra $X \subset \mathbb{R}^n$, $Y \subset \mathbb{R}^m$ such that $X \times Y$ contains the feasible region G of Problem (P). Such polyhedra can be constructed by standard methods of convex programming (see, e.g. [4, 25]) if S, in addition, is compact (very often in practical problems). Let f_* denote the optimal value of (P) (we always adopt the convention that an optimal value equals ∞ if no feasible points exist).

Given two convex polyhedra $B \subset Y$ and $T \supset S$ we define Problem $P(B, T)$ as

$$\min\{f(x, y) \colon x \in X,\ y \in B,\ g(x, y) \leq 0,\ (x, y) \in T\}\ ,$$

and its relaxed problem as

$$R(B, T) \quad \min\{f(x, y) \colon x \in X,\ y \in B,\ g(x, y) \leq 0,\ u \in B,\ (x, u) \in T\}\ .$$

By $\beta(B, T)$ we denote the optimal value of $R(B, T)$. Due to our compactness assumption, whenever $\beta(B, T) < \infty$ then $R(B, T)$ has an optimal solution. By α_{k-1} we shall denote the least upper bound for f_* known at the beginning of iteration k.

Roughly speaking the algorithm runs as follows: At a typical step k, say, we are given a polyhedral convex set $T_k \supset S$ and a collection Γ_k of subsets $B \subset Y$, all described by affine inequalities, such that any optimal solution of (P) is contained in $X \times \cup\{B \colon B \in \Gamma_k\}$. Since $\beta(B, T_k)$ cannot be higher than the optimal value of $P(B, T_k)$ this implies that $f_* \geq \min\{\beta(B, T_k) \colon B \in \Gamma_k\}$. For all $B \in \Gamma_k$ the relaxed problem $R(B, T_k)$ should be solved. If $\beta(B, T_k) \geq \alpha_k$, then B is deleted from Γ_k. Out of the remaining sets one, say B_k, is selected such that $\beta(B_k, T_k) = \min\{\beta(B, T_k) \colon B \in \Gamma_k\}$. Now either we cut off part of T_k, thus obtaining a subset T_{k+1} such that $S \subset T_{k+1}$, or we bisect B_k, thus obtaining two complementary subsets B_k^+ and B_k^- of B_k. We replace B_k by $\{B_k^+, B_k^-\}$ to obtain Γ_{k+1}. The algorithm may terminate finitely; it will do so in particular if $f_* = \infty$.

For describing the method we need a continuous function $\psi(x, y, u)$ defined on $X \times Y \times \mathbb{R}^m$ such that

$$\psi(x, y, \cdot\,) \text{ is convex,}$$
$$\psi(x, y, y) = 0\ ,$$
$$\psi(x, y, u) \leq 0 \Rightarrow \begin{cases} f(x, u) \leq f(x, y)\ , \\ g(x, u) \leq g(x, y)\ . \end{cases}$$

Then in iteration k, given $(x^k, y^k, u^k) \in X \times Y \times Y$, we shall use a function $l_k(\cdot)$ defined on $\mathbb{R}^m$ by

$$\left.\begin{aligned} l_k(v) &:= \langle t^k, v - y^k \rangle , \\ t^k \in \partial\psi_k(u^k),\ \psi_k(u) &:= \psi(x^k, y^k, u) . \end{aligned}\right\} \tag{1}$$

The following properties (only) of $l_k(\cdot)$ will be utilized:

(i) $l_k(\cdot)$ is affine. If $l_k(u^k) \leq 0$, then $\psi(x^k, y^k, u^k) \leq 0$.
(ii) $l_k(y^k) = 0$.
(iii) There exists a constant M such that $\|\nabla l_k\| \leq M$ for all k.
(iv) If $(x^k, y^k, u^k) \to (x, y, u)$ and $\limsup_{k\to\infty} l_k(u^k) \leq 0$, then $f(x, u) \leq f(x, y)$, $g(x, u) \leq g(x, y)$.

Property (i) follows from

$$\begin{aligned} l_k(u^k) = \langle t^k, u^k - y^k \rangle &\geq \psi(x^k, y^k, u^k) - \psi(x^k, y^k, y^k) \\ &= \psi(x^k, y^k, u^k) . \end{aligned}$$

Property (ii) is obvious.
Property (iii) follows from

$$\begin{aligned} \|\nabla l_k\| = \|t^k\| &= \langle t^k, t^k / \|t^k\| \rangle \\ &\leq \psi(x^k, y^k, u^k + t^k/\|t^k\|) - \psi(x^k, y^k, u^k) \\ &\leq \text{const.}, \end{aligned}$$

the last inequality being due to the fact that ψ is continuous and the arguments of ψ remain within a compact set. To verify (iv) we use once more that

$$l_k(u^k) \geq \psi(x^k, y^k, u^k) ,$$

and continuity of ψ. Then from the hypothesis of (iv) follows $\psi(x, y, u) \leq 0$, and therefore $f(x, u) \leq f(x, y)$, $g(x, u) \leq g(x, y)$.

Simple examples for ψ and l_k will be given in paragraph 3 below. The algorithm can now be described in detail.

ALGORITHM

Initialization. With the two convex polyhedra $B_0 := Y$ and $T_0 \supset S$ (for instance $T_0 := \mathbb{R}^n \times \mathbb{R}^m$, or $T_0 := S$ if S is a polyhedral convex set) solve Problem $R(B_0, T_0)$. If $\beta(B_0, T_0) = \infty$, terminate: Problem (P) has no feasible point.
If $\beta(B_0, T_0) < \infty$, let $(x^{B_0}, y^{B_0}, u^{B_0})$ be the obtained optimal solution of $R(B_0, T_0)$. Let $\alpha_{-1} := \infty$ and $\Gamma_0 := \{B_0\}$.

Iteration k $(k = 0, 1, \ldots)$. At the beginning of iteration k we have a collection Γ_k of convex polyhedral subsets $B \subset B_0$ such that every solution of (P) is contained in $X \times \bigcup\{B\colon B \in \Gamma_k\}$, and we have a polyhedron $T_k \supset S$. For each $B \in \Gamma_k$ we know $\beta(B, T_k)$ and, if $\beta(B, T_k) < \infty$, we know an optimal solution (x^B, y^B, u^B) of $R(B, T_k)$. Furthermore $\alpha_{k-1} \geq f_*$ is at hand.
Let

$$F_k := \{(x^B, v^B)\colon v^B \in \{y^B, u^B\},\ (x^B, v^B) \in G,\ B \in \Gamma_k\}\ ,$$
$$\alpha_k := \min\{\alpha_{k-1},\ \min\{f(x, u) : (x, u) \in F_k\}\}$$

(the currently known smallest upper bound for f_*), and, if $\alpha_k < \infty$, let (ξ^k, η^k) be the best feasible point known so far so that $f(\xi^k, \eta^k) = \alpha_k$. Let

$$\Delta_k := \{B \in \Gamma_k : \beta(B, T_k) \leq \alpha_k\}\ .$$

Select $B_k \in \Delta_k$ such that

$$\beta(B_k, T_k) = \min\{\beta(B, T_k) : B \in \Delta_k\}\ .$$

Let $\beta_k := \beta(B_k, T_k)$.

(1) If $\beta_k \geq \alpha_k$, terminate: $f_* = \alpha_k$. If $\alpha_k < \infty$, then (ξ^k, η^k) is an optimal solution of (P) and if $\alpha_k = \infty$, then (P) has no feasible point.[1]

(2) If $\beta_k < \alpha_k$, then let $(x^k, y^k, u^k) := (x^{B_k}, y^{B_k}, u^{B_k})$, and let $l_k(\cdot)$ be given according to (1).

[Note that $\beta_k = f(x^k, y^k)$.]

(2a) If $l_k(u^k) \leq 0$, then set $\Gamma_{k+1} := \Delta_k$ and go to (3).

[Note that in this case $(x^k, u^k) \notin S$, because otherwise from (i) (x^k, u^k) is feasible for (P) and one has $f(x^k, u^k) \leq f(x^k, y^k) = \beta_k$. Hence $\alpha_k \leq f(x^k, u^k) \leq \beta_k$, which contradicts the hypothesis of (2).]

(2b) If $l_k(u^k) > 0$, then set $c_k := l_k(u^k)/2$ and let

$$B_k^- := \{y \in B_k : l_k(y) \leq c_k\}\ ,\quad B_k^+ := \{y \in B_k : l_k(y) \geq c_k\}\ .$$

[Note that $y^k \in B_k^-$, since $l_k(y^k) = 0$ from property (ii) of l_k, and $u^k \in B_k^+$.]
Let

$$\Gamma_{k+1} := (\Delta_k \setminus \{B_k\}) \cup \{B_k^-, B_k^+\}\ .$$

(3) Set

[1] Alternatively we may use strict inequality in the definition of Δ_k and terminate if $\Delta_k = \emptyset$. The case $\alpha_k = f_*$ and no termination must then be treated separately (the remaining sets may no longer contain a solution).

$$T_{k+1} := \begin{cases} T_k, \text{ if } (x^k, u^k) \in S\,, \\ \{(x, u) \in T_k : \varphi(x^k, u^k) + t_1^T(x - x^k) + t_2^T(u - u^k) \leq 0\} & \text{otherwise}\,, \end{cases}$$

where (t_1, t_2) is a subgradient of φ at (x^k, u^k).
For each $B \in \Gamma_{k+1}$ solve $R(B, T_{k+1})$. Go to iteration $k + 1$.[2]
This completes the description of the algorithm.

COMMENT. The just described algorithm can be considered as a combination of branch-and-bound and cutting plane methods. If the variable y is missing in Problem (P), then the algorithm becomes a pure cutting plane method. If $T_0 = S$ (and thereby $T_k = S$ for all k), then the algorithm becomes a pure branch-and-bound method.

A crucial operation in the above algorithm is the solution of the relaxed Problem $R(B, T_k)$. This question will be discussed below for some special cases of Problem (P). Now we are going to give some examples for separation functions $l_k(\cdot)$ satisfying properties (i)–(iv).

3. Examples for the Separation Function

(1) Let Φ: $\mathbb{R}^m \to \mathbb{R}$ be a convex (hence continuous) function such that $\Phi(0) = 0$ and $\Phi(u) > 0$ for all $u \neq 0$ (for instance $\Phi(u) = \|u\|$ or $\Phi(u) = \|u\|^2$).
Let

$$\psi(x, y, u) := \Phi(u - y)\,.$$

Then ψ has all the properties requested. For instance, $\psi(x, y, u) \leq 0 \Rightarrow u = y \Rightarrow f(x, u) \leq f(x, y)$ etc. Hence ψ can be used to define the separation function $l_k(\cdot)$ by means of (1).

(2) Assume that $f(x, \cdot)$ and $g(x, \cdot)$ are convex. Let

$$\psi(x, y, u) := \max\{f(x, u) - f(x, y),\, g(x, u) - g(x, y)\}\,.$$

The ψ has the properties requested and so can be used to define $l_k(\cdot)$ by means of (1).

(3) In the following example $l_k(\cdot)$ is not determined through (1). But $l_k(\cdot)$ will still possess properties (i)–(iv) which are enough for performing the algorithm.

Let $\psi(x, y, u)$ be as in example (1) or – if $f(x, \cdot)$ and $g(x, \cdot)$ are convex – as in

[2] If $T_{k+1} = T_k$ we solve $R(B, T_{k+1})$ for each $B \in \Gamma_{k+1} \backslash \Gamma_k$, since for $B \in \Gamma_k$ we have already solved $R(B, T_k)$.

example (2). Let $\psi_k(u) := \psi(x^k, y^k, u)$. Select $t \in \partial\psi_k(u^k)$. Select $j0 \in \{1, \dots, m\}$ such that

$$t_{j0}(u^k - y^k)_{j0} = \max_j t_j(u^k - y^k)_j .$$

Then define $l_k(v) := t_{j0}(v - y^k)_{j0}$. It follows that

$$\begin{aligned} l_k(u^k) = t_{j0}(u^k - y^k)_{j0} &= \max_j t_j(u^k - y^k)_j \geqslant \frac{1}{m} \langle t, u^k - y^k \rangle \\ &\geqslant \frac{1}{m} (\psi_k(u^k) - \psi_k(y^k)) = \frac{1}{m} \psi(x^k, y^k, u^k) . \end{aligned}$$

Therefore $l_k(u^k) \leqslant 0 \Rightarrow \psi(x^k, y^k, u^k) \leqslant 0$. From this and

$$\|\nabla l_k\| = \|(0, \dots, t_{j0}, \dots, 0)\| \leqslant \|t\| ,$$

it follows again that properties (i)–(iv) are fulfilled.

This choice of $l_k(\cdot)$ is particularly practical, since it leads to rectangular subdivisions. In fact, the hyperplane

$$\{v \,|\, l_k(v) = l_k(u^k)/2\} ,$$

which splits B_k into B_k^+ and B_k^-, becomes now

$$\{v \,|\, v_{j0} = \tfrac{1}{2}(u^k + y^k)_{j0}\} .$$

So, if B_k is a rectangle, B_k^+ and B_k^- are rectangles, too.

In certain cases to be considered below the subproblems $R(B, T_k)$ call for examining the vertices of B, and this task is greatly facilitated if the B are rectangles.

4. Convergence of the Algorithm

We turn now to the convergence of the above algorithm. From the construction of T_k, it follows that $S \subset T_{k+1} \subset T_k$ for every k. This implies $\beta_k \leqslant \beta_{k+1} \leqslant f_*$ for all k. Hence $\beta_* := \lim \beta_k$ exists and $\beta_* \leqslant f_*$. If the algorithm terminates at iteration k, i.e., $\beta_k \geqslant \alpha_k$, then from $\alpha_k \geqslant f_*$ follows $\alpha_k = \beta_k = f_*$. If the algorithm does not terminate then we have the following convergence result:

THEOREM. (a) *$\beta_k \nearrow f_*$, and $\{(x^k, u^k)\}$ has a limit point which solves (P).*

(b) *If (x^k, u^k) is feasible for almost all k, then $\alpha_k \searrow f_*$, and every limit point of $\{(\xi^k, \eta^k)\}$ solves (P).*

Proof. (a) If $\beta_k = \infty$ for some k, then the algorithm terminates at iteration k and $f_* = \infty$ (Problem (P) has no feasible point). Thus if the algorithm does not terminate, then $\beta_k < \infty$ for all k. Since $\beta_k = \beta(B_k, T_k) < \infty$, Problem $R(B_k, T_k)$ has an optimal solution (x^k, y^k, u^k). Note that $(x^k, u^k) \in T_k$ for every k. This and the rule for constructing T_k imply, by a standard argument (e.g. [4], p. 240), that any limit point of $\{(x^k, u^k)\}$ belongs to S. We distinguish two cases.

Case 1. The case 2b occurs only finitely often. In this case we may assume without loss of generality that case 2a occurs for all k. From case 2a follows $l_k(u^k) \leq 0$ for all k, hence $\limsup l_k(u^k) \leq 0$. Let (x^*, y^*, u^*) be any limit point of $\{(x^k, y^k, u^k)\}$. Then $g(x^*, y^*) \leq 0$. Hence we obtain from property (iv) of l_k that $f(x^*, u^*) \leq f(x^*, y^*)$ and $g(x^*, u^*) \leq 0$. Since $f(x^k, y^k) = \beta_k \leq f_*$ this implies $f(x^*, u^*) \leq f_*$. Since $(x^*, u^*) \in S$, (x^*, u^*) is feasible, and therefore $f(x^*, u^*) = f_*$. This and the monotonicity of $\{\beta_k\}$ imply that $\beta_k \nearrow f_*$.

Case 2. The case 2b occurs infinitely often. In this case there exists a nested subsequence of $\{B_k\}$ which for the simple notation we also denote by $\{B_k\}$. By extracting subsequences if necessary, we may assume that $(x^k, y^k, u^k) \to (x^*, y^*, u^*)$.

If $B_{k+1} \subset B_k^-$, then in particular $u^{k+1} \in B_k^-$, which implies

$$l_k(u^{k+1}) \leq c_k = l_k(u^k)/2 .$$

Hence

$$0 \leq l_k(u^k)/2 \leq l_k(u^k) - l_k(u^{k+1}) \leq M\|u^k - u^{k+1}\| .$$

If $B_{k+1} \subset B_k^+$, then in particular $y^{k+1} \in B_k^+$, which implies

$$l_k(y^{k+1}) \geq c_k = l_k(u^k)/2 .$$

Since $l_k(y^k) = 0$ it follows that

$$0 \leq l_k(u^k)/2 \leq l_k(y^{k+1}) - l_k(y^k) \leq M\|y^{k+1} - y^k\| .$$

Hence always $l_k(u^k) \to 0$. From property (iv) it follows that $f(x^*, u^*) \leq f(x^*, y^*)$ and $g(x^*, u^*) \leq 0$. Since (x^*, u^*) is feasible the claim follows as in the previous case.

(b) Assume now that (x^k, u^k) is feasible for all k large enough. From part (a) we see that $\{(x^k, u^k)\}$ has a limit point (x^*, u^*) which solves (P). Since $f_* \leq \alpha_k = f(\xi_k, \eta_k) \leq f(x^k, u^k)$ for every k large enough, and $\{\alpha_k\}$ is nonincreasing, it follows that any limit point of $\{(\xi^k, \eta^k)\}$ solves (P), and $\alpha_k \searrow f_*$. The theorem is proved. □

REMARK. If $g(x, y) = g(x)$ is independent of y and $T_0 = S$, then (x^k, u^k) is feasible for every k. In fact in this case Problem $R(B, S)$ reads

$$\min\{f(x, y): x \in X,\ y \in B,\ u \in B,\ g(x) \leq 0,\ (x, u) \in S\} .$$

5. Subproblems

The solution of $R(B, T_k)$ is crucial for implementing the algorithm. Here we collect some special cases where the subproblem $R(B, T_k)$ can be solved, at least in principle, by available methods.

(1) Assume that $g(x, y) = g(x)$ is independent of y, that g is quasiconvex and that $f(x, y)$ is convex in x and quasiconcave in y. Then from quasiconcavity follows

$$\min\{f(x, y): y \in B\} = \min_i f(x, v^i),$$

where v^i are the vertices of B. Therefore

$$\begin{aligned}\beta(B, T_k) &= \min\{f(x, y): x \in X, \ y \in B, \ g(x) \leq 0, \ u \in B, \ (x, u) \in T_k\} \\ &= \min\{\min_i f(x, v^i): x \in X, \ g(x) \leq 0, \ u \in B, \ (x, u) \in T_k\} \\ &= \min_i \min\{f(x, v^i): x \in X, \ g(x) \leq 0, \ u \in B, \ (x, u) \in T_k\}.\end{aligned}$$

Hence the subproblem $R(B, T_k)$ is reduced to finitely many convex programs (minimizing a convex function over a convex set), one for each vertex v^i. $R(B, T_k)$ simplifies further if $f(x, y) = f_1(x) + f_2(y)$; then

$$\beta(B, T_k) = \min\{f_1(x): x \in X, \ g(x) \leq 0, \ u \in B, \ (x, u) \in T_k\} + \min_i f_2(v^i),$$

a single convex program combined with an examination of the vertices of B. Since B is generated from some predecessor B' by adding one affine inequality, the vertices of B could be calculated (for small m at least) from those of B' by some available methods, see, e.g. [13]. Of course, the vertices of B are determined explicitly if B is a rectangle – see example 3 above for the separation function. Finally if B is a rectangle, $B := \{y \in \mathbb{R}^m: a \leq y \leq b\}$, and if $f_2(y)$ is separable, $f_2(y) := \Sigma_j f_{2j}(y_j)$ with f_{2j} a quasiconcave function of one variable, then

$$\min\{f_2(y): y \in B\} = \sum_j \min\{f_{2j}(a_j), \ f_{2j}(b_j)\}.$$

(2) Assume now that $f(x, y) = f(x)$ is independent of y, that f is convex and that $g(x, y)$ is quasiconvex in x and quasiconcave in y. Then from quasiconcavity follows

$$\min\{g(x, y): y \in B\} = \min_i g(x, v^i),$$

where v^i are the vertices of B, and therefore

$$\begin{aligned}\beta(B, T_k) &= \min\{f(x): x \in X, \ y \in B, \ g(x, y) \leq 0, \ u \in B, \ (x, u) \in T_k\} \\ &= \min\{f(x): x \in X, \ \min_i g(x, v^i) \leq 0, \ u \in B, \ (x, u) \in T_k\} \\ &= \min_i \min\{f(x): x \in X, \ g(x, v^i) \leq 0, \ u \in B, \ (x, u) \in T_k\}.\end{aligned}$$

Hence for each v^i it is required to minimize a convex function over a convex set. This simplifies further if $g(x, y) := g_1(x) + g_2(y)$. Then

$$\beta(B, T_k) = \min\{f(x) \colon x \in X,\ g_1(x) + \gamma \leq 0,\ u \in B,\ (x, u) \in T_k\}$$

with $\gamma := \min_i g_2(v^i) = \min\{g_2(y) \colon y \in B\}$.

REMARK. One may replace the branching rule used in the algorithm by a nonadaptive branching rule which ensures that all sets B are simplices, provided B_0 is a simplex. Namely, if B_k is a simplex, then let v^k, w^k be the vertices of B_k such that the edge $[v^k, w^k]$ is longest among the edges of B_k. Then B_k^- and B_k^+ are obtained from B_k by replacing v^k and w^k respectively by the midpoint of the edge $[v^k, w^k]$. This bisection has the property that any *nested* subsequence of the simplices generated by it contracts to a single point (see, e.g. [12, 22, 29]). With this simplex bisection we have the following result:

$$\beta_k \nearrow f_*, \text{ and each limit point of } \{(x^k, u^k)\} \text{ solves } (P)\,.$$

To see this, let $\{(x^{k(j)}, y^{k(j)}, u^{k(j)})\}_{j\in\mathbb{N}}$ be a subsequence of $\{(x^k, y^k, u^k)\}$ converging to (x^*, y^*, u^*). Then there exists a nested subsequence $\{B_{k(q)}\}_{q\in\mathbb{N}}$ of $\{B_k\}$ and a subsequence $\{B_{k(j(q))}\}_{q\in\mathbb{N}}$ of $\{B_{k(j)}\}$ such that $B_{k(j(q))} \subset B_{k(q)}$ for all q (see [22] for details). Consequently $y^{k(j(q))}$, $u^{k(j(q))} \in B_{k(q)}$ for all q. Since the nested subsequence $B_{k(q)}$ contracts to a point $\bar{u}$, say, it follows that $y^* = u^* = \bar{u}$. The claim follows then by the same argument as in the convergence theorem above.

6. An Example

We illustrate one possible realization of our method by the following example:

$$\min f(x, y) := \left(\frac{x_1}{2} - x_2\right) \frac{2y_1 - 1}{y_2 + 1}$$

$$\text{subject to } (x, y) \in S := \{(x, y) \in \mathbb{R}^4_+ \,|\, \varphi(x, y) := x_1^2 + x_2^2 + y_1^2 + y_2^2 - 1 \leq 0\}.$$

For solving this problem we shall use the rectangle branching (example 3 in the paper).

Initialization: Take $T_0 = \{(x, y) \in \mathbb{R}^4_+ \,|\, x_1 + x_2 + y_1 + y_2 - 2 \leq 0\}$, $B_0 = [0, 1] \times [0, 1]$. Then $T_0 \supset S$. The relaxed problem $R(B_0, T_0)$ reads

$$\min\left\{\left(\frac{x_1}{2} - x_2\right) \frac{2y_1 - 1}{y_2 + 1} \,\Big|\, x_1 + x_2 \leq 1,\ y \in B_0,\ u \in B_0,\ (x, u) \in T_0\right\}$$

$$= \min_{y \in V(B_0)} \beta(y, T_0)\,,$$

where $V(B_0)$ denotes the vertex set of B_0, and

$$\beta(y, T_0) := \min\left\{\left(\frac{x_1}{2} - x_2\right)\frac{2y_1 - 1}{y_2 + 1} \,|x_1 + x_2 \leq 1,\ u \in B_0,\ (x, u) \in T_0\right\}.$$

$V(B_0)$ has four vertices $(0,0)$, $(0,1)$, $(1,0)$ and $(1,1)$.
At $y = (0,0)$ we have $(x_1/2 - x_2)(-1/1) = x_2 - x_1/2$. Thus

$$\beta(y, T_0) = \min\left\{\left(x_2 - \frac{x_1}{2}\right) |x_1 + x_2 \leq 1,\ u \in B_0,\ (x, u) \in T_0\right\}.$$

This linear problem has an optimal solution $x = (1,0)$, $u = (0,0)$.
Hence $\beta(y, T_0) = -\frac{1}{2}$.
At $y = (0,1)$, $\beta(y, T_0) = -\frac{1}{4}$ with $x = (1,0)$, $u = (0,0)$.
At $y = (1,0)$ we have

$$\beta(y, T_0) = \min\left\{\left(\frac{x_1}{2} - x_2\right) |x_1 + x_2 \leq 1,\ u \in B_0,\ (x, u) \in T_0\right\} = -1$$

with $x = (0,1)$, $u = (1,0)$.
At $y = (1,1)$, $\beta(y, T_0) = -\frac{1}{2}$ with $x = (0,1)$, $u = (1,0)$.
Hence $\beta(B_0, T_0) := \min_{y \in V(B_0)} \beta(y, T_0) = -1$ with $x = (0,1)$, $u = (1,0)$, $y = (1,0)$. Set $\Gamma_0 = \{B_0\}$.

Iteration $k = 0$: $\alpha_0 = f(1,0,0,0) = -\frac{1}{2}$ and $(\xi^0, \eta^0) = (1,0,0,0)$. $\Delta_0 := \{B \in \Gamma_0 | \beta(B, T_0) \leq -\frac{1}{2}\} = \{B_0\} = \Gamma_0$. Thus $\beta_0 = -1$ with $x^0 = (0,1)$, $u^0 = (1,0)$, $y^0 = (1,0)$.

We take $l_0(v) := l_{j0}(v - y^0)_{j0}$ as in example 3 of the paper with $t = \mathrm{d}/\mathrm{d}u \|u - y_0\|^2|_{u=u_0}$.
Thus

$$l_0(u^0) = 0 \text{ (since } y^0 = u^0\text{)},$$

and therefore we arrive at case (2a) of the algorithm. Hence $\Gamma_1 = \Delta_0 = \{B_0\}$. Since $(x^0, u^0) \notin S$ we have

$$\begin{aligned} T_1 &= \left\{(x, u) \in T_0 | \varphi(x^0, u^0) + \frac{\mathrm{d}\varphi(x^0, u^0)}{\mathrm{d}x}(x - x^0)\right. \\ &\qquad \left. + \frac{\mathrm{d}\varphi(x^0, u^0)}{\mathrm{d}u}(u - u_0) \leq 0\right\} \\ &= \{(x, u) \in T_0 | 2x_2 + 2u_1 - 3 \leq 0\}. \end{aligned}$$

We compute $\beta(B, T_1)$ by solving

$$\min_{y \in V(B_0)} \beta(y, T_1)$$

where

$$\beta(y, T_1) := \min\left\{\left(\frac{x_1}{2} - x_2\right)\frac{2y_1 - 1}{y_2 + 1} \,|x_1 + x_2 \leq 1,\ u \in B_0,\ (x, u) \in T_1\right\}.$$

For each $y \in V(B_0)$ we compute $\beta(y, T_1)$ and we get the following:
For $y = (0, 0)$, $\beta(y, T_1) = -\frac{1}{2}$ with $x = (1, 0)$, $u = (0, 0)$.
For $y = (1, 0)$, $\beta(y, T_1) = -1$ with $x = (0, 1)$, $u = (0, 0)$.
For $y = (1, 1)$, $\beta(y, T_1) = -\frac{1}{4}$ with $x = (0, 1)$, $u = (0, 0)$.
For $y = (0, 1)$, $\beta(y, T_1) = -\frac{1}{4}$ with $x = (1, 0)$, $u = (0, 0)$.
(Note that $\beta(y, T_1) = f(x, y)$)
Hence $\beta(B_0, T_1) = -1$ with $x = (0, 1)$, $u = (0, 0)$ and $y = (1, 0)$.

Iteration $k = 1$: $T_1 := \{(x, u)|x_1 + x_2 + u_1 + u_2 - 2 \leq 0, 2x_2 + 2u_1 - 3 \leq 0, x_i, u_i \geq 0\ (i = 1, 2)\}$, $\Gamma_1 = \{B_0\}$, $\beta(B_0, T_1) = -1$, $x^1 = (0, 1)$, $u^1 = (0, 0)$, $y^1 = (1, 0)$. Best upper bound $\alpha_1 = -\frac{1}{2}$; best feasible point $(\xi^1, \eta^1) = (1, 0, 0, 0)$.
Thus $\Delta_1 = \Gamma_1 = \{B_0\}$, $\beta_1 = -1$, $B_1 = B_0$.
Since $\alpha_1 > \beta_1$ and $l_1(u^1) = t_{j0}(u^1 - y^1)_{j0} = 2(j_0 = 1)$ we go to case (2b) of iteration k.
Then $B_1^+ = [0, \frac{1}{2}] \times [0, 1]$, $B_1^- = [\frac{1}{2}, 1] \times [0, 1]$, $\Gamma_2 = \{B_1^+, B_1^-\}$.
Since $(x^1, u^1) \in S$ we have $T_2 = T_1$.
We have to compute $\beta(B_1^+, T_2)$ and $\beta(B_1^-, T_2)$.

$$\beta(B_1^+, T_2) := \min_{y \in V(B_1^+)} \min\left\{\left(\frac{x_1}{2} - x_2\right)\frac{2y_1 - 1}{y_2 + 1} \,|x_1 + x_2 \leq 1,\right.$$
$$\left.(x, u) \in T_2,\ u \in B_1^+\right\}.$$

$$V(B_1^+) = \{(0, 0), (\tfrac{1}{2}, 0), (\tfrac{1}{2}, 1), (0, 1)\}.$$

For each $y \in V(B_1^+)$ we solve the linear program:

$$\beta(y, T_2) := \min\left\{\left(\frac{x_1}{2} - x_2\right)\frac{2y_1 - 1}{y_2 + 1} \,|x_1 + x_2 \leq 1,\ (x, u) \in T_2,\ u \in B_1^+\right\}.$$

For $y = (0, 0)$, $\beta(y, T_2) = -\frac{1}{2}$ with $x = (1, 0)$, $u = (0, 0)$.
For $y = (\frac{1}{2}, 0)$, $\beta(y, T_2) = 0$ with $x = (1, 0)$, $u = (\frac{1}{2}, 0)$.
For $y = (\frac{1}{2}, 1)$, $\beta(y, T_2) = 0$ with $x = (1, 0)$, $u = (\frac{1}{2}, 0)$.
For $y = (0, 1)$, $\beta(y, T_2) = -\frac{1}{4}$ with $x = (1, 0)$, $u = (0, 0)$.
Hence $\beta(B_1^+, T_2) = \min_{y \in V(B_1^+)} \beta(y, T_2) = -\frac{1}{2}$.
To compute $\beta(B_1^-, T_2)$ we solve, for each fixed $y \in V(B_1^-)$, the linear program:

$$\beta(y, T_2) := \min\left\{\left(\frac{x_1}{2} - x_2\right)\frac{2y_1 - 1}{y_2 + 1} \,|x_1 + x_2 \leq 1,\ (x, u) \in T_2,\ u \in B_1^-\right\}.$$

We get $\beta(B_1^-, T_2) = -1$ at $x = (0, 1)$, $u = (\frac{1}{2}, 0)$, $y = (1, 0)$.

Iteration $k=2$: $\Gamma_2=\{B_1^+, B_1^-\}$, $\beta(B_1^+, T_2)=-\frac{1}{2}$, $\beta(B_1^-, T_2)=-1$, $x^2=(0,1)$, $u^2=(\frac{1}{2},0)$, $y^2=(1,0)$; $\alpha_2=-\frac{1}{2}$, $(\xi^2,\eta^2)=(1,0,0,0)$, $\Delta_2=\Gamma_2$, $B_2=B_1^-$, $\beta_2=-1$. Since $\alpha_2>\beta_2$ and $l_2(u^2)=\max_j t_j(u^2-y^2)_j=1$ we go to case (2b), and bisect B_2 into B_2^+ and B_2^-: $B_2^+=[\frac{1}{2}, \frac{3}{4}]\times[0,1]$, $B_2^-=[\frac{3}{4}, 1]\times[0,1]$. Then $\Gamma_3=\{B_1^+, B_2^+, B_2^-\}$. Since $(x^2, u^2)=(0,1,\frac{1}{2},0)\not\in S$ we have

$$T_3=\left\{(x,u)\in T_2|\varphi(x^2,u^2)+\frac{d\varphi}{dx}(x^2,u^2)(x-x^2)+\frac{d\varphi}{du}(x^2,u^2)(u-u^2)\le 0\right\}$$
$$=\left\{(x,u)\in T_2|2x_2+u_1-\frac{9}{4}\le 0\right\}.$$

For each $B\in\Gamma_3$ we compute $\beta(B, T_3)$. Since $\beta(B_1^+, T_2)=-\frac{1}{2}=\alpha_2$, the set B_1^+ is deleted from further consideration. To compute $\beta(B_2^+, T_3)$ we solve

$$\min_{y\in V(B_2^+)}\min\left\{\left(\frac{x_1}{2}-x_2\right)\frac{2y_1-1}{y_2+1}\,|(x,u)\in T_3,\ x_1+x_2\le 1,\ u\in B_2^+\right\}.$$

For each $y\in V(B_2^+)$ fixed, we solve the linear program

$$\min\left\{\left(\frac{x_1}{2}-x_2\right)\frac{2y_1-1}{y_2+1}\,|(x,u)\in T_3,\ x_1+x_2\le 1,\ u\in B_2^+\right\},$$

obtain $\beta(B_2^+, T_3)=-\frac{7}{16}$ with the corresponding solution $x=(0,\frac{7}{8})$, $u=(\frac{1}{2},0)$, $y=(\frac{3}{4},0)$. Similarly $\beta(B_2^-, T_3)=-\frac{3}{4}$ with $x=(0,\frac{3}{4})$, $u=(\frac{3}{4},0)$, $y=(1,0)$.

Iteration $k=3$: $\Gamma_3=\{B_1^+, B_2^+, B_2^-\}$, $\beta(B_1^+, T_3)=-\frac{1}{2}$, $\beta(B_2^+, T_3)=-\frac{7}{16}$, $\beta(B_2^-, T_3)=-\frac{3}{4}$. Hence $B_3=B_2^-$, $\beta_3=-\frac{3}{4}$, $\alpha_3=\alpha_2=-\frac{1}{2}$, $(\xi^3,\eta^3)=(1,0,0,0)$; $x^3=(0,\frac{3}{4})$, $u^3=(\frac{3}{4},0)$, $y^3=(1,0)$. We terminate the algorithm and obtain an approximate solution $(\xi^3,\eta^3)=(1,0,0,0)$ with $f(\xi^3,\eta^3)=-\frac{1}{2}$. Since $\beta_3=-\frac{3}{4}$ is a lower bound for the optimal value f_* we have $0<f(\xi^3,\eta^3)-f_*\le -\frac{1}{2}+\frac{3}{4}=\frac{1}{4}$. Hence (ξ^3,η^3) is an ε-optimal solution with $\varepsilon:=\frac{1}{4}$.

References

1. Al-Khayyal, F. A. and Falk, J. E. (1983), Jointly constrained biconvex programming, *Mathematics of Operations Research* **8**, 273–286.
2. Benders, J. F. (1962), Partitioning procedures for solving mixed-variables programming problems, *Numerische Mathematik* **4**, 238–252.
3. Benson, H. P. (1982), On the convergence of two branch-and-bound algorithms for nonconvex programming problems, *Journal of Optimization Theory and Applications* **36**, 129–134.
4. Blum, E. and Oettli, W. (1975), *Mathematische Optimierung*, Springer, Berlin.
5. Cheney, E. W. and Goldstein, A. A. (1959), Newton's method for convex programming and Tchebycheff approximation, *Numerische Mathematik* **1**, 253–268.
6. Dantzig, G. B. and Wolfe, P. (1960), Decomposition principle for linear programs, *Operations Research* **8**, 101–111.

7. Falk, J. E. and Soland, R. M. (1969), An algorithm for separable nonconvex programming problems, *Management Science* **15**, 550–569.
8. Fukushima, M. (1983), An outer approximation algorithm for solving general convex programs, *Operations Research* **31**, 101–113.
9. Fukushima, M. (1984), On the convergence of a class of outer approximation algorithms for convex programs, *Journal of Computational and Applied Mathematics* **10**, 147–156.
10. Fukushima, M. (1984), A descent algorithm for nonsmooth convex optimization, *Mathematical Programming* **30**, 163–175.
11. Geoffrion, A. (1970), Elements of large-scale mathematical programming, Part I: Concepts, *Management Science* **16**, 652–675.
12. Horst, R. (1976), An algorithm for nonconvex programming problems, *Mathematical Programming* **10**, 312–321.
13. Horst, R., de Vries, J., and Thoai, N. V. (1988), On finding new vertices and redundant constraints in cutting plane algorithms for global optimization, *Operations Research Letters* **7**, 85–90.
14. Horst, R. (1988), Deterministic global optimization with partition sets whose feasibility is not known. Application to concave minimization, reverse convex constraints, DC-programming, and Lipschitzian optimization, *Journal of Optimization Theory and Applications* **58**, 11–37.
15. Horst, R. and Tuy, H. (1987), On the convergence of global methods in multiextremal optimization, *Journal of Optimization Theory and Applications* **54**, 253–271.
16. Horst, R. and Tuy, H. (1990), *Global Optimization*, Springer, Berlin; second edition 1992.
17. Kelley, J. E. (1960), The cutting-plane method for solving convex programs, *Journal of the SIAM* **8**, 703–712.
18. Kiwiel, K. C. (1985), *Methods of Descent for Nondifferentiable Optimization*, Lecture Notes in Mathematics **1133**, Springer, Berlin.
19. Mayne, D. Q. and Polak, E. (1984), Outer approximation algorithm for nondifferentiable optimization problems, *Journal of Optimization Theory and Applications* **42**, 19–30.
20. McCormick, G. P. (1976), Computability of global solutions to factorable nonconvex programs: Part I – Convex underestimating problems, *Mathematical Programming* **10**, 147–175.
21. Muu, L. D. (1985), A convergent algorithm for solving linear programs with an additional reverse convex constraint, *Kybernetika (Prague)* **21**, 428–435.
22. Muu, L. D. and Oettli, W. (1991), An algorithm for indefinite quadratic programming with convex constraints, *Operations Research Letters* **10**, 323–327.
23. Muu, L. D. and Oettli, W. (1991), A method for minimizing a convex-concave function over a convex set, *Journal of Optimization Theory and Applications* **70**, 377–384.
24. Pardalos, P. M. (1987), Generation of large-scale quadratic programs for use as global optimization test problems, *ACM Transactions on Mathematical Software* **13**, 133–137.
25. Pardalos, P. M. and Rosen, J. B. (1987), *Constrained Global Optimization: Algorithms and Applications*. Lecture Notes in Computer Science **268**.
26. Rosen, J. B. and Pardalos, P. M. (1986), Global minimization of large-scale constrained concave quadratic problems by separable programming, *Mathematical Programming* **34**, 163–174.
27. Soland, R. M. (1971), An algorithm for separable nonconvex programming problems II: Nonconvex constraints, *Management Science* **17**, 759–773.
28. Topkis, D. M. (1970), Cutting-plane methods without nested constraint sets, *Operations Research* **18**, 404–413.
29. Thoai, N. V. and Tuy, H. (1980), Convergent algorithms for minimizing a concave function, *Mathematics of Operations Research* **5**, 556–566.
30. Tuy, H. (1987), Global minimization of a difference of two convex functions, *Mathematical Programming Study* **30**, 150–182.
31. Tuy, H. and Horst, R. (1988), Convergence and restart in branch-and-bound algorithms for global optimization. Application to concave minimization and d.c. optimization problems, *Mathematical Programming* **41**, 161–183.
32. Tuy, H., Thieu, T. V., and Thai, N. Q. (1985), A conical algorithm for globally minimizing a concave function over a closed convex set, *Mathematics of Operations Research* **10**, 498–514.
33. Veinott, A. F. (1967), The supporting hyperplane method for unimodal programming, *Operations Research* **15**, 147–152.

Addendum

Since the first submission of this manuscript several other publications combining branch-and-bound with outer approximation have appeared. We mention the following.

For concave minimization:

Benson, H. P. and Horst, R.: A Branch-and-Bound-Outer Approximation Algorithm for Concave Minimization over a Convex Set, *J. Computers and Mathematics with Applications* **21**, 67–76 (1991).

Horst, R., Thoai, N. V., and Benson, H. P.: Concave Minimization via Conical Partitions and Polyhedral Outer Approximation, *Mathematical Programming* **50**, 259–274 (1991).

Horst, R., Thoai, N. V., and de Vries, J.: A New Simplicial Cover Technique in Constrained Global Optimization, *J. Global Optimization* **2**, 1–19 (1992).

Horst, R., Thoai, N. V., and de Vries, J.: On Geometry and Convergence of a Class of Simplicial Covers, *Optimization* **25**, 53–64 (1992).

For d.c.-programs:

Horst, R., Phong, T. Q., and Thoai, N. V.: On Solving General Reverse Convex Programming Problems by a Sequence of Linear Programs and Line Searches, *Annals of Operations Research* **25**, 1–18 (1990).

Horst, R., Phong, T. Q., Thoai, N. V., and de Vries, J.: On Solving a DC Programming Problem by a Sequence of Linear Programs, *J. Global Optimization* **1**, 183–203 (1991).