J Glob Optim (2010) 47:485–501
DOI 10.1007/s10898-009-9475-2

$\mathcal{D}(\mathcal{C})$-optimization and robust global optimization

Hoang Tuy

Received: 21 August 2009 / Accepted: 25 September 2009 / Published online: 11 October 2009

Abstract For solving global optimization problems with nonconvex feasible sets existing methods compute an approximate optimal solution which is not guaranteed to be close, within a given tolerance, to the actual optimal solution, nor even to be feasible. To overcome these limitations, a robust solution approach is proposed that can be applied to a wide class of problems called $\mathcal{D}(\mathcal{C})$-optimization problems. DC optimization and monotonic optimization are particular cases of $\mathcal{D}(\mathcal{C})$-optimization, so this class includes virtually every nonconvex global optimization problem of interest. The approach is a refinement and extension of an earlier version proposed for dc and monotonic optimization.

Keywords Nonconvex global optimization · Approximate optimal solution · Robust approach · Essential optimal solution · dc optimization · dm (monotonic) optimization · $\mathcal{D}(\mathcal{C})$-optimization · Successive Incumbent Transcending Algorithm

AMS Subjcet Classfication 90C26 · 90C30 · 90C31 · 90C57 · 49K40 · 65K05

1 Introduction

Consider the general nonconvex global optimization problem

$$\min\{f(x)|\ g_i(x) \le 0\ (i = 1, \dots, m),\ x \in [a, b]\}, \tag{P}$$

where $f, g_1, \dots, g_m$ are nonconvex continuous real-valued functions on $\mathbb{R}^n,\ a, b \in \mathbb{R}^n$, and $[a, b] := \{x \in \mathbb{R}^n|\ a \le x \le b\}$.

In the most general case, a common approach for solving these problems is to use a suitable outer approximation or branch and bound procedure for generating a sequence of infeasible solutions x^k such that, as $k \to +\infty,\ x^k$ tends to a feasible solution x^*, while $f(x^k)$ tends

H. Tuy (✉)
Institute of Mathematics, 18 Hoang Quoc Viet, Hanoi 10307, Vietnam
e-mail: htuy@math.ac.vn

to the optimal value of the problem. Under these conditions x^* is a global optimal solution, so theoretically the problem can be considered solved.

In practical computation, however, the algorithm has to be stopped at some x^k expected to be close enough to the optimal solution. Since $f(x^k)$ is a lower bound for the optimal value, if a feasible solution y^k is available such that $\|y^k - x^k\| \to 0$, then, by continuity, $f(y^k) - f(x^k) \to 0$, so, given a tolerance $\eta > 0$ one can determine k such that $f(y^k) - f(x^k) \leq \eta$. The feasible solution y^k which satisfies $f(y^k) \leq \min(P) + \eta$ is called an *η-optimal solution*.

Unfortunately, for hardest nonconvex problems finding a feasible solution y^k is almost as difficult as solving the problem itself. There is then no simple way to determine how large k should be taken to guarantee an error no more than η if $f(x^k)$ is accepted as the optimal value. Furthermore, if the global optimal solution happens to be an isolated feasible point (which is not uncommon), then even if an η-optimal solution can be computed effectively, it may change drastically upon a small change of η, causing serious difficulties for practical implementation.

For example, when $f(x)$, $g_i(x)$, $i = 1, \ldots, m$, are polynomials on $\mathbb{R}^n$, Lasserre's method [3] for solving (P) defines, for each $k = 1, 2, \ldots,$ a finite set A_k of n-vectors with nonnegative integral components together with a set of additional variables $y = (y_\alpha) \in \mathbb{R}^{|A_k|}$ and a semidefinite program SDP_k in the variables (x, y), such that the problem (P) is equivalent to SDP_k with the additional nonconvex constraints

$$y_\alpha = x^\alpha, \quad \alpha \in A_k. \tag{1}$$

Then SDP_k is a relaxation of (P), as it is obtained from a problem equivalent to (P) by omitting the nonconvex constraints (1). Therefore, if ω_k is the optimal value of SDP_k, then $\omega_k \leq \min(P)$ and with an appropriately chosen sequence A_k, $k = 1, 2, \ldots,$ it can be proved that

$$\omega_k \nearrow \min(P) \quad (k \to +\infty). \tag{2}$$

Thus, for k satisfying $\omega_k \geq \min(P) - \eta$, the value ω_k yields an approximate optimal value of (P) with tolerance η. The trouble, however, is that, for a given $\eta > 0$ we do not know how large k should be taken. Moreover, if (x^k, y^k) is an optimal solution of the relaxed problem SDP_k then generally $\max_{\alpha \in A_k} |(y^k)_\alpha - (x^k)^\alpha| > 0$ and so x^k is infeasible with a significant feasibility error, even for large values of k. Not to mention the difficulty of implementation if the optimal solution happens to be an isolated feasible solution.

Another well known method—the RLT method [4]—builds, for each $k = 1, 2, \ldots,$ a linear program LP_k in the variables (x, y) such that (P) is equivalent to LP_k with the additional constraints (1). So LP_k is a linear relaxation of (P) with optimal value $\omega_k \leq \min(P)$ and though (2) does not generally hold, with a suitable choice of A_k this bound ω_k can be incorporated into a branch and bound algorithm to generate a sequence $\{(x^{(k)}, y^{(k)})\}$ such that

$$\max_{\alpha \in \mathcal{A}_k} |[y^{(k)}]_\alpha - [x^{(k)}]^\alpha| \to 0 \ (k \to +\infty).$$

Under these conditions $\bar{x} = \lim_{k \to +\infty} x^{(k)}$ is an optimal solution of (P), but again the trouble is that for a given tolerance $\eta > 0$ we do not know when to stop to have a guaranteed η-approximate optimal value. Also, $x^{(k)}$ is infeasible and, due to the huge dimension of (x^k, y^k), the feasibility error may be significant even for large values of k. Moreover, just as in Lasserre's method, the error cannot be controlled if the optimal solution happens to be an isolated point of the feasible set.

In some practical implementations of the RLT approach (see e.g. [1]), if an η-optimal solution $(\bar{x}, \bar{y})$ of LP$_k$ satisfies $\max_{\alpha\in\mathcal{A}_k} |\bar{y}_\alpha - \bar{x}^\alpha| \leq \varepsilon$ then $\bar{x}$ is called an (ε, η)-*optimal solution* of the problem (P). For k sufficiently large, the RLT method provides an (ε, η)-optimal solution, but, as was shown in [13], this concept of (ε, η)-optimal solution may not be quite adequate, since such a solution may not lie in the proximity of the exact optimal solution.

To sum up, most current methods for solving (P), as ingenious as they may be, are devised for finding only an approximate optimal value but not an approximate optimal solution. In the general case, they do not provide a guaranteed upper bound for the error of approximation, i.e. a guaranteed lower bound of the degree of accuracy of the solution.

In an attempt to overcome these limitations, a robust approach to nonconvex optimization was developed in [12], and specialized with some improvement to polynomial programming in [13]. More recently, this approach has been applied successfully to quadratic optimization under quadratic constraints [15]. In the present paper we intend to refine the method and extend it to a wide class of problems that includes dc , dm and virtually every global optimization problem of interest.

After the Introduction, in Sect. 2 we discuss some basic concepts of branch and bound algorithms for nonconvex global optimization. In Sect. 3 we introduce $\mathcal{D}(\mathcal{C})$-optimization as the most general class of global optimization problems that, in our view, can be reasonably proposed for study by deterministic methods. This class includes dc optimization and dm optimization as two most important subclasses. In Sect. 4 a robust approach is presented for $\mathcal{D}(\mathcal{C})$-optimization. In Sect. 5 the method is specialized to dc and dm optimization. The SIT algorithms to be presented in this section contain several improvements upon the earlier versions in [12,13,15]. Section 6 presents some illustrative examples and Sect. 7 concludes the paper with some remarks.

2 Remarks on BB algorithms for global optimization

2.1 Problems with hard feasible set

Consider a nonconvex problem (P) where the feasible set

$$D := \{x \in [a, b] \mid g_i(x) \leq 0,\ i = 1, \ldots, m\}$$

is such that computing just one feasible solution may be very hard. To solve such a problem a common method is to generate a sequence of boxes (hyperrectangles) $M_k \subset [a, b]$ together with a sequence $\beta(M_k) \in \mathbb{R} \cup \{+\infty\},\ k = 1, 2, \ldots,$ such that $M_1 = [a, b]$ and

$$\text{diam} M_k \to 0 \text{ as } k \to +\infty, \tag{3}$$

$$\beta(M_k) \leq \inf\{f(x) \mid x \in M_k \cap D\}, \tag{4}$$

$$\beta(M_k) \leq \min(P), \tag{5}$$

$$\beta(M_k) < +\infty \Rightarrow M_k \cap D \neq \emptyset. \tag{6}$$

These properties are usually achieved through a BB (branch and bound) procedure, involving two suitably defined basic operations: 1) Bounding: for every partition set M, compute a lower bound $\beta(M)$ for $f(x)$ over the feasible solutions in M; 2) Branching: at each iteration k, select a box M_k among all partition sets still of interest for exploration and subdivide it into two or more subboxes.

Condition (3) says that the subdivision process is exhaustive, while (4) indicates that $\beta(\mathrm{M}_k)$ is a lower bound of $f(x)$ over the feasible solutions in M_k, and (5) follows from the fact that M_k has smallest lower bound among all partition sets currently of interest. As for (6) it is a requirement that bounds are assumed to satisfy in order to ensure that

$$M_k \cap D \neq \emptyset \quad \forall k. \tag{7}$$

(in view of (5), if $M_k \cap D = \emptyset$ then (6) implies that min(P)$= +\infty$, i.e. (P) is infeasible)

2.1.1 Bounding

A lower bounding method $M \mapsto \beta(M)$ is said to be *valid* if it satisfies (4) and (6) [hence, also (5) and (7)]. Condition (6) is essential, failing which $M_k \cap D$ may be empty for some k, so that M_{k_ν} may shrink to an infeasible solution. Several published BB algorithms are incorrect just because of the failure of condition (6).

Due to condition (6), at iteration k any M with $\beta(M) = +\infty$ is deleted. (Since D is assumed to be very hard, no feasible solution—a fortiori no current best feasible solution—is available at any iteration, so $\beta(M) = +\infty$ is the unique pruning criterion). Let $\mathcal{R}_k$ denote the collection of boxes that remain after pruning. If $\mathcal{R}_k = \emptyset$ the algorithm stops: the problem is infeasible. Otherwise, a partition set $M_k \in \mathcal{R}_k$ with $\beta(M_k) = \min\{\beta(M)|\ M \in \mathcal{R}_k\}$ is selected for further partitioning.

A common practice for computing a valid lower bound is to take an underestimator $\varphi_M(x)$ of $f(x)$ over M, i.e. a function satisfying $\varphi_M(x) \leq f(x)\ \forall x \in M$ and setting

$$\beta(M) = \inf\{\varphi_M(x)|\ x \in M \cap \Omega\}, \tag{8}$$

where $\Omega \supset D$ is selected so that the problem (8) can be solved efficiently and, moreover, condition (6), and hence, (7), is satisfied.

Another natural condition that, preferably, bounds should satisfy is that $\beta(M') \geq \beta(M)$ whenever $M' \subset M$. If this is so, to obtain tight bounds one may try, before computing $\beta(M)$, to *reduce* the box M i.e. to replace it by a smaller box $M' \subset M$ without losing any feasible point in M. BB algorithms using such reduction operations before bounding are referred to as *BRB (Branch-Reduce-and-Bound) algorithms.*

2.1.2 Branching

A popular subdivision method for BB algorithms is the standard rectangular bisection which consists in dividing each M_k into two equal smaller boxes by a hyperplane orthogonal to the longest edge of M_k at its midpoint. As is well known (see e.g. [7]), this subdivision method is *exhaustive*, i.e., whenever infinite it ensures the existence of a filter (nested sequence of boxes) M_{k_ν} satisfying $\mathrm{diam} M_{k_\nu} \to 0\ (\nu \to +\infty)$ [condition (3)].

Proposition 1 *If lower bounds are consistent with branching, i.e.*

$$\beta(M) - \min\{f(x)|\ x \in M \cap D\} \to 0 \ \text{ as } \mathrm{diam} M \to 0, \tag{9}$$

then $\beta(M_k) \to \min$ (P) $(k \to +\infty)$ (the algorithm is *convergent*).

Proof From (3) for every k there must exist $x^k \in M_k \cap D$. Let $z^k \in M_k \cap D$ be a minimizer of $f(x)$ over $M_k \cap D$ (the points x^k and z^k exist, but may not be known effectively). Since the subdivision rule is exhaustive there exists a filter $\{M_{k_\nu}\}$ such that diam$M_{k_\nu} \to 0$. We then have $z^{k_\nu} - x^{k_\nu} \to 0$, so z^{k_ν} and x^{k_ν} tend to a common limit $x^* \in D$. But by (9), $f(z^{k_\nu}) - \beta(M_{k_\nu}) \to 0$, i.e. $\beta(M_{k_\nu}) \to f(x^*)$, hence, in view of (5), $f(x^*) \leq$ min(P). Since $x^* \in D$ the latter implies $f(x^*) =$ min(P), i.e. $\beta(M_{k_\nu}) \to$ min(P)). □

The above proof highlights the key role of condition (6): if it does not hold, then $M_{k\nu} \cap D$ may be empty for some ν, so x^* is infeasible, implying failure of the algorithm: $\beta(M_{k_\nu}) \not\to$ min(P), despite (9). Also note that since x^k may not be known effectively, the algorithm does not, in general, provide a sequence of feasible solutions converging to an optimal solution.

2.1.3 (ε, η)-*optimal solution*

In practice the algorithm must be stopped at some iteration when a sufficiently good approximation of x^* can be obtained.

Given the tolerances $\varepsilon > 0,\ \eta > 0$, an $x \in \mathbb{R}^n$ satisfying $g_i(x) \leq \varepsilon,\ i = 1, \ldots, m$ is referred to as an ε-*approximate solution* and an ε-approximate solution $\bar{x}$ such that $f(\bar{x}) \leq$ min (P) $+ \eta$ is called an (ε, η)-*approximate optimal solution*.

Proposition 2 *For given* $\varepsilon > 0,\ \eta > 0$ *one can determine k such that* diamM_k *is sufficiently small to ensure that any point* $z \in M_k$ *yields an* (ε, η)-*approximate optimal solution.*

Proof Denote by (P_k) the problem (P) restricted to M_k, i.e. $\min\{f(x) | x \in M_k \cap D\}$, and let z^k be an optimal solution of (P_k). By (9) min $(P_{k_\nu}) - \beta(M_{k_\nu}) \to 0$, hence, $f(z^{k_\nu}) - \beta(M_{k_\nu}) \to 0$. Since $\beta(M_{k_\nu}) \leq$ min(P) $\leq$ min$(P_{k_\nu}) = f(z^{k_\nu})$, it follows that $f(z^{k_\nu}) -$ min (P) $\to 0$. Since, on the other hand, $z^{k_\nu} \in M_{k_\nu}$ and $g_i(z^{k_\nu}) \leq 0$ $(i = 1, \ldots, m)$, while diam$M_{k_\nu} \to 0$, there exists ν such that any $\bar{x} \in M_{k_\nu}$ satisfies $g_i(\bar{x}) \leq \varepsilon$ $(i = 1, \ldots, m)$ and $f(\bar{x}) -$ min (P) $\leq \eta$. Then $\bar{x}$ is an (ε, η)-approximate optimal solution. □

This proposition indicates that both the RLT method and the BB version of Lasserre's method can be refined to provide a guaranteed (ε, η)-approximate optimal solution. Clearly this concept of approximate optimal solution is more appropriate than that of (ε, η)-optimal solution used in [1], which is defined as an ε-approximate solution $\bar{x}$ such that $f(\bar{x}) \leq f(x) + \eta$ for all ε-approximate solutions x. In fact, while an (ε, η)-optimal solution may lie far away from the true optimum [12,15], an (ε, η)-approximate optimal solution in the above defined sense is always close to the optimum. Still the difficulty remains, because, as will be seen from an example in Sect. 4, if (P) happens to have an optimal solution $\bar{x}$ which is an isolated feasible solution then a slight change of the data (or the tolerances $\varepsilon > 0, \eta > 0$) may cause a drastic change of the (ε, η)-approximate optimal solution.

2.2 Problems with nice feasible set

The above analysis points out the difficulties of BB algorithms for solving (P) when the feasible set is nonconvex: convergence must be guaranteed by an exhaustive subdivision process, very few partition sets can be pruned at each iteration, while an approximate solution can be reached only when diamM_k becomes very small. In brief, BB algorithms for solving problems with a nonconvex constraint set are very slow and can at most produce (ε, η)-approximate optimal solutions.

By contrast, we now show that many of these difficulties can be avoided or at least alleviated when the constraint set D is *nice*, i.e. such that a feasible point can be computed at cheap cost. In fact, in this case, at each iteration a current best feasible solution CBS exists which is the best among all feasible solutions so far available, thus allowing every partition set M with $\beta(M)$ exceeding the objective function value at CBS to be pruned. Furthermore, a feasible solution approximating the optimal solution can be obtained instead of an infeasible solution approximating an optimal solution within certain tolerances. Finally an adaptive subdivision process can be used which may ensure a faster convergence than an exhaustive subdivision process.

2.2.1 Adaptive subdivision and η-optimal solution

A feasible solution x^* of (P) is called an *η-optimal solution* if it satisfies $f(x^*) \leq \min\,(\mathrm{P}) + \eta$.

When the feasible set D is nice, a BB algorithm can produce an η-optimal solution in finitely many steps, while its convergence can be sped up by using an *adaptive subdivision process* instead of an exhaustive one.

Specifically suppose for every M_k two points x^k, y^k in M_k can be constructed such that

$$x^k \in M_k \cap D, \quad y^k \in M_k, \tag{10}$$

$$f(y^k) - \beta(M_k) = o(\|x^k - y^k\|). \tag{11}$$

For instance, these conditions hold if $\beta(M_k) = \min\{\varphi_{M_k}(x) |\ x \in M_k \cap D\}$, [see (8)], where $\varphi_{M_k}(x)$ is an underestimator of $f(x)$ tight at $y^k \in M_k$ (i.e. satisfying $\varphi_{M_k}(x^k) = f(y^k)$), while $x^k \in \mathrm{argmin}\{\varphi_{M_k}(x) |\ x \in M_k \cap D\}$; they also hold if $f(x) = f_1(x) - f_2(x)$, $x^k \in \mathrm{argmin}\{f_1(x) |\ x \in M_k \cap D\}$, $y^k \in \mathrm{argmax}\{f_2(x) |\ x \in M_k\}$, $\beta(M_k) = f_1(x^k) - f_2(y^k)$. In Sect. 5 we will discuss in more detail how to construct x^k, y^k satisfying conditions (10), (11). Under these conditions, the following *adaptive bisection* can be used to further subdivide M_k (see [7], p. 160):

Bisect M_k by the hyperplane $x_s = (y^k_s + x^k_s)/2$, where $s \in \mathrm{argmax}_{i=1,\dots,n}|y^k_i - x^k_i|$.

Proposition 3 *A BB algorithm using an adaptive bisection rule is convergent.*

Proof It is well known (see e.g. [7]) that the adaptive bisection ensures the existence of a subsequence $\{k_\nu\}$ such that

$$y^{k_\nu} - x^{k_\nu} \to 0\ (\nu \to +\infty) \tag{12}$$

$$\lim_{\nu\to+\infty} y^{k_\nu} = \lim_{\nu\to+\infty} x^{k_\nu} = x^* \in D. \tag{13}$$

From (11) and (13) we deduce $\beta(M_{k_\nu}) \to f(x^*)$ as $\nu \to +\infty$. But, since $x^* \in D$, we have $\beta(M_{k_\nu}) \leq f(x^*)$. Therefore, $\beta(M_{k_\nu}) \to \min(P)$ $(\nu \to +\infty)$, proving the convergence of the algorithm. □

Observe that the standard bisection can be considered as a special case of adaptive subdivision when $[x^k, y^k] = M_k$. The condition (11) which ensures the convergence of the BB algorithm then becomes the convergence condition (9) in Proposition 1.

We shall refer to a BB algorithm using an adaptive (exhaustive, resp.) subdivision rule as an *adaptive* (*exhaustive* , resp.) BB algorithm.

Proposition 4 *A convergent adaptive* BB *algorithm yields an η-optimal solution in finitely many steps. Specifically, x^k is η-optimal when $f(x^k) - \beta(M_k) \leq \eta$.*

Proof We have $f(x^{k_\nu}) - \beta(M_{k_\nu}) \leq f(x^{k_\nu}) - \min\,(\mathrm{P}) \to 0$, whence the result. □

To sum up, global optimization problems with nonconvexity concentrated in the objective function are generally easier to handle by BB algorithm than those with nonconvex constraints. This suggests that in dealing with nonconvex global optimization problems one should try to use valid transformations to shift, in one way or another, all the nonconvexity in the constraints to the objective function.

3 $\mathcal{D}(\mathcal{C})$-optimization

For a global optimization problem to be tractable by deterministic methods it must have some mathematical structure. It is by analyzing this structure that one can get insight into relevant properties guiding the search for the optimum. In this section we describe a general mathematical structure which offers a convenient framework for a robust approach to global optimization.

For any two functions $g_1, g_2 : \mathbb{R}^n \to \mathbb{R}$ write $g = g_1 \vee g_2,\ h = g_1 \wedge g_2$ if $g(x) = \max\{g_1(x), g_2(x)\},\ h(x) = \min\{g_1(x), g_2(x)\}$.

Let $\mathcal{C}$ be a family of real-valued functions on $\mathbb{R}^n$ such that

(i) $f, g \in \mathcal{C},\ \alpha, \beta \in \mathbb{R}_+ \Rightarrow \alpha f + \beta g \in \mathcal{C}$;
(ii) $g_1, g_2 \in \mathcal{C} \Rightarrow g_1 \vee g_2 \in \mathcal{C}$.
Occasionally, when it is necessary to specify that we are considering functions on $\mathbb{R}^n$ we write $\mathcal{C}_n$.

Proposition 5 *Under assumptions* (i) and (ii) *the family* $\mathcal{D}(\mathcal{C}) = \mathcal{C} - \mathcal{C}$ *is a vector lattice with respect to the two operations* $\vee$ *and* $\wedge$.

Proof From (i) and $\mathcal{D}(\mathcal{C}) = \mathcal{C} - \mathcal{C}$ it follows that $\mathcal{D}(\mathcal{C})$ is a vector space. If $f = f_1 - f_2, g = g_1 - g_2$ with $f_1, f_2, g_1, g_2 \in \mathcal{C}$ then $\max\{f, g\} = \max\{f_1 + g_2, g_1 + f_2\} - (f_2 + g_2) \in \mathcal{D}(\mathcal{C})$ because $f_1 + g_2 \in \mathcal{C},\ g_1 + f_2 \in \mathcal{C},\ f_2 + g_2 \in \mathcal{C}$. This proves that $f, g \in \mathcal{D}(\mathcal{C}) \Rightarrow f \vee g \in \mathcal{D}(\mathcal{C})$. Furthermore, $\min\{f, g\} = -\max\{-f, -g\}$, hence $f \wedge g \in \mathcal{D}(\mathcal{C})$. □

Also note that if $f \in \mathcal{D}(\mathcal{C})$ then $|f| \in \mathcal{D}(\mathcal{C})$ because $|f| = \max\{f, -f\}$.

An optimization problem of the form (P) where $f, g_i \in \mathcal{D}(\mathcal{C}), i = 1, \ldots, m$, is called a $\mathcal{D}(\mathcal{C})$-*optimization problem* on $\mathbb{R}^n$.

Assume that, in addition to (i), (ii), $\mathcal{C}$ satisfies

(iii) Every function $x \mapsto x_i$, with $i \in \{1, \ldots, n\}$, belongs to $\mathcal{C}$.

Two most important cases when $\mathcal{C}$ satisfies (i), (ii), (iii) are:

(1) $\mathcal{C}$ is the family of convex functions. Any $f \in \mathcal{D}(\mathcal{C})$ is then called a *dc function* and a $\mathcal{D}(\mathcal{C})$-optimization problem is called a *dc optimization* problem. Until recently most problems studied in global optimization can be shown to belong to this class, see [2,7].
(2) $\mathcal{C}$ is the family of increasing functions, i.e. functions $f(x)$ such that $x' \leq x \Rightarrow f(x') \leq f(x)$. Any $f \in \mathcal{F}$ is then called a *dm function* (difference of monotonic function) and a $\mathcal{D}(\mathcal{C})$-optimization problem is called a *dm optimization*, or else a *monotonic optimization* problem. A theory of monotonic optimization has emerged in recent years that has been shown to parallel dc optimization in several respects ([8,15], also [10,9], [11,14]).

Since any polynomial can be viewed either as a dc or a dm function, the set of dc functions or dm functions on a box $[a, b]$ is dense in the space $C[a, b]$ of continuous functions on $[a, b]$ with the supnorm topology. Virtually every global optimization of interest belongs to either of the above described basic classes.

Proposition 6 *Assume $\mathcal{C}$ satisfies* (i),(ii), (iii). *Then any $\mathcal{D}(\mathcal{C})$-optimization problem can be rewritten in the standard form*

$$\min\{f(x)|\ g_1(x) - g_2(x) \geq 0,\ x \in [a, b]\}, \quad \textit{where } f, g_1, g_2 \in \mathcal{C}. \tag{14}$$

Proof Problem (P), where $f = f_1 - f_2$ with $f_1, f_2 \in \mathcal{C}$, can be written as a problem in $z = (x, t) \in \mathbb{R}^{n+1}$ as follows:

$$\min\{f_1(x) + t|\ f_2(x) + t \geq 0,\ g_i(x) \geq 0\ (i = 1, \ldots, m),\ x \in [a, b],\ t \in [\underline{t}, \bar{t}]\}$$

where $\underline{t} = -f_2(b),\ \bar{t} = -f_2(a)$. The objective function now is $f_1(x) + t \in \mathcal{C}_{n+1}$ (because $h \in \mathcal{C}_{n+1}$ if $h : (x, t) \mapsto t$), while the set of constraints $f_2(x) + t \geq 0,\ g_i(x) \geq 0\ (i = 1, \ldots, m)$ can be replaced by a single constraint $g(x, t) \geq 0$, with $g(x, t) := \min\{f_2(x) + t, g_1(x), \ldots, g_m(x)\} \in \mathcal{D}(\mathcal{C}_{n+1})$. Changing the notation yields (14). □

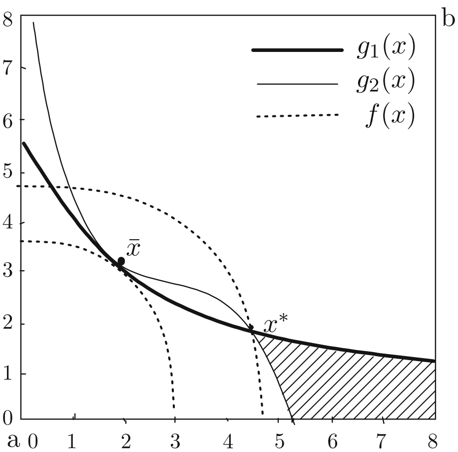
Corollary 1 *Any dc (dm , resp.) optimization problem can be written in the form (14) where f, g_1, g_2 are convex (increasing, resp.) functions.*

4 Robust approach

A $\mathcal{D}(\mathcal{C})$-optimization problem can obviously be solved by a BB algorithm as discussed in Sect. 2.1. By Proposition 2, for given tolerances $\varepsilon > 0, \eta > 0$, it is possible to obtain by such an algorithm a guaranteed (ε, η)-optimal solution. However, this concept of approximate optimality is not quite adequate when the problem happens to have an isolated optimal solution—a situation not quite uncommon in nonconvex global optimization. The example depicted in Fig. 1 shows that an (ε, η)-optimal solution may change drastically upon a slight change of the tolerances $\varepsilon > 0,\ \eta > 0$.

In this problem the objective function is $f(x)$ and the constraint set consists of points x satisfying $g_1(x) \leq 0,\ g_2(x) \leq 0,\ x \in [a, b]$. The global optimal solution $\bar{x}$ is an isolated feasible point. If $g_2(x)$ is perturbed to $g_2(x) + \delta$ where $\delta > 0$ is very small then the global optimal solution moves to x^* which lies far away from $\bar{x}$. So a slight perturbation of the data may cause a drastic change of the global optimal solution; consequently, a small change of the tolerances $\varepsilon > 0, \eta > 0$ may cause a drastic change of the (ε, η)-optimal solution.

Fig. 1 Isolated optimal solution



Because of this instability an isolated optimal solution is very difficult to compute and very difficult to implement when computable. To avoid these difficulties most global optimization algorithms assume that the feasible set D is *robust*, i.e. satisfies

$$D = \text{cl}(\text{int})D,$$

where ClA, int$A$ denote the closure and the interior, respectively, of the set A. However, in turn this condition is generally very hard to check for a given nonconvex problem. Practically we often have to consider feasible sets which are not known a priori to contain isolated points or not.

Therefore, from a practical point of view a method is highly desirable which could help to discard isolated feasible solutions from consideration without having to check their presence. To this end it is convenient to introduce some concepts.

A nonisolated feasible solution x^* of (P) is called an *essential optimal solution* if

$$f(x^*) \leq f(x) \quad \forall x \in D^*,$$

where D^* denotes the set of nonisolated points (accumulation points) of $D := \{x \in [a, b] | \; g(x) \leq 0\}$. In other words, an essential optimal solution is an optimal solution of the problem

$$\min\{f(x) | \; x \in D^*\}.$$

For given $\varepsilon > 0$, $\eta > 0$, a point $x \in [a, b]$ satisfying $g(x) \leq \varepsilon$ is said to be ε-*feasible*, and a nonisolated feasible solution x^* is called an *essential* (ε, η)-*optimal solution* if

$$f(x^*) \leq f(x) + \eta \quad \forall x \in D_\varepsilon,$$

where $D_\varepsilon := \{x \in [a, b] | \; g(x) \leq \varepsilon\}$ is the set of all ε-feasible solutions.

Clearly for $\varepsilon = \eta = 0$ an essential (ε, η)-optimal solution is a nonisolated feasible point which is optimal.

The above discussion suggests that instead of trying to find an optimal solution to (P), it would be more practical and reasonable to look for an essential (ε, η)-optimal solution.

Below we present a solution approach embodying this point of view for $\mathcal{D}(\mathcal{C})$-optimization, i.e. for a class of problems that includes virtually every nonconvex global optimization problem of interest.

4.1 Interchangeability between objective and constraint

From now on we consider problem (P) where $g, g_1, \ldots, g_m \in \mathcal{D}(\mathcal{C})$. Setting $g(x) = \min_{i=1,\ldots,m} g_i(x)$, we rewrite (P) as

$$\min\{f(x) | \; g(x) \leq 0, \; x \in [a, b]\}, \tag{P}$$

where $f, g \in \mathcal{D}(\mathcal{C})$. Further, by Proposition 6, without loss of generality we can assume $f \in \mathcal{C}$. Given $\varepsilon, \gamma \in \mathbb{R}$, let us consider the pair of optimization problems

$$\min\{f(x) | \; g(x) \leq \varepsilon, \; x \in [a, b]\}, \tag{P$_\varepsilon$}$$

$$\min\{g(x) | \; f(x) \leq \gamma, \; x \in [a, b]\}, \tag{Q$_\gamma$}$$

where the objective and constraint functions are interchanged. Due to the fact $f \in \mathcal{C}$, a key feature of problem (Q$_\gamma$) for our purpose is that its feasible set is a *convex* set (if $\mathcal{C}$ is the set of convex functions) or a *normal* set (if $\mathcal{C}$ is the set of increasing functions, see [8]), so in either case problem (Q_γ) has no isolated feasible solution and computing a feasible solution to (Q_γ) can be done at cheap cost Therefore, as was shown in Sect. 2.2, an adaptive BB algorithm can be devised that, for any given tolerance $\eta > 0$, can compute an η-optimal solution of (Q_γ) in finitely many steps (Proposition 4). Furthermore, such an η-optimal solution is stable under small changes of η.

For our purpose of solving (P), this property of (Q_γ) combined with the following 'reciprocity' relationship between the two problems (P_ε) and (Q_γ) [5,6] turned out to be crucial.

Proposition 7 *For every* $\varepsilon > 0$, *if* $\min(Q_\gamma) > \varepsilon$ *then* $\min(P_\varepsilon) > \gamma$.

Proof If $\min(Q_\gamma) > \varepsilon$ then any $x \in [a, b]$ such that $g(x) \le \varepsilon$ must satisfy $f(x) > \gamma$, hence, by compactness of the feasible set of (P_ε), $\min\{f(x)|\ g(x) \le \varepsilon, x \in [a, b]\} > \gamma$.

□

Proposition 7 simply expresses the interchangeability (duality) between objective and constraint in many practical situations [6]. Exploiting this interchangeability, the robust approach to $\mathcal{D}(\mathcal{C})$-optimization to be presented below consists in replacing the original problem (P), possibly very difficult, by a sequence of easier, stable, problems (Q_γ), where the parameter γ can be iteratively adjusted until a stable (robust) solution to (P) is obtained.

4.2 Successive incumbent transcending

A key step towards finding a global optimal solution of a problem (P) is to deal with the following question of incumbent transcending.

$(*, \gamma)$ *Given a real number* γ, *and* $\varepsilon > 0$, *check whether problem* (P) *has a nonisolated feasible solution* x *satisfying* $f(x) \le \gamma$, *or else establish that no* ε*-feasible solution* x *exists such that* $f(x) \le \gamma$.

Using Proposition 7, consider an adaptive BB algorithm for solving problem (Q_γ). As described in Sect. 2, such an algorithm generates at each iteration k two points $x^k \in M_k,\ y^k \in M_k$ satisfying [see (10, 11)]:

$$f(x^k) \le \gamma, \quad g(y^k) - \beta(M_k) \to 0 \quad \text{as } k \to +\infty,$$

where $\beta(M_k) \le \min(Q_\gamma)$ (Note that the objective function is $g(x)$, the constraints are $f(x) \le \gamma,\ x \in [a, b]$)

Proposition 8 *Let* $\varepsilon > 0$ *be given. Either* $g(x^k) < 0$ *for some* k *or* $\beta(M_k) > \varepsilon$ *for some* k. *In the former case,* x^k *is a nonisolated feasible solution of* (P) *satisfying* $f(x^k) \le \gamma$. *In the latter case, no* ε*-feasible solution* x *of* (P) *exists such that* $f(x) \le \gamma$ *(so, if* $\gamma = f(\bar{x}) - \eta$ *for a given* $\eta > 0$ *and* $\bar{x}$ *is a nonisolated feasible solution, then* $\bar{x}$ *is an essential* (ε, η)*-optimal solution of* (P)*).*

Proof If $g(x^k) < 0$ then x^k is an essential feasible solution of (P) satisfying $f(x^k) \le \gamma$, because by continuity $g(x) < 0$ for all x in a neighborhood of x^k. If $\beta(M_k) > \varepsilon$ then min $(Q_\gamma) > \varepsilon$, hence, by Proposition 7, $\min(P_\varepsilon) > \gamma$, i.e. $f(\bar{x}) \le \min(P_\varepsilon) + \eta$.

It remains to show that either $g(x^k) < 0$ or $\alpha(M_k) > \varepsilon$ for some k. Suppose the contrary, that

$$g(x^k) \ge 0, \quad \alpha(M_k) \le \varepsilon \quad \forall k. \tag{15}$$

As we saw in Sect. 2, the sequence x^k, y^k generated by an adaptive BB algorithm satisfies, for some infinite subsequence $\{k_\nu\} \subset \{1, 2, \ldots\}$,

$$g(y^{k_\nu}) - \alpha(M_{k_\nu}) \to 0 \ (\nu \to +\infty), \tag{16}$$

$$\lim_{\nu\to+\infty} x^{k_\nu} = \lim_{\nu\to+\infty} y^{k_\nu} = x^* \quad \text{with } f(x^*) \leq \gamma,\ x^* \in [a, b]. \tag{17}$$

Since by (15) $g(x^k) \geq 0\ \forall k$, it follows that $g(x^*) \geq 0$. On the other hand, by (16) and (15) we have $g(x^*) = \lim_{\nu\to+\infty} \beta(M_{k_\nu}) \leq \varepsilon$. This contradiction shows that the event (15) is impossible, completing the proof. □

Thus, with the stopping criterium "$g(x^k) < 0$ or $\beta(M_k) > \varepsilon$" an adaptive BB procedure for solving (Q_γ) will help to solve the subproblem (*, γ). Note that $\beta(M_k) = \max\{\alpha(M)|\ M \in \mathcal{R}_k\}$ ($\mathcal{R}_k$ being the collection of partition sets remaining for exploration at iteration k). So if the deletion (pruning) criterion in each iteration of this procedure is "$\beta(M) > \varepsilon$" then the stopping criterion "$\beta(M_k) > \varepsilon$" is nothing but the usual criterion "$\mathcal{R}_k = \emptyset$".

For brevity an adaptive BB algorithm for solving (Q_γ) with the deletion criterion "$\beta(M) > \varepsilon$" and stopping criterion "$g(x^k) < 0$" will be referred to as *Procedure* (*, γ).

Using this procedure, problem (P) can be solved according to the following scheme, where γ_0 denotes an arbitrary real number larger than $\max\{f(x)|\ x \in [a, b]\}$.

Successive Incumbent Transcending Scheme

Start with $\gamma = \gamma_0$.

Call Procedure (*, γ). If a nonisolated feasible solution $\bar{x}$ of (P) is obtained with $f(\bar{x}) \leq \gamma$, reset $\gamma \leftarrow f(\bar{x}) - \eta$ and repeat. Otherwise, stop: $\bar{x}$ is an essential (ε, η)-optimal solution if $\gamma = f(\bar{x}) - \eta$; problem (P) has no ε-feasible solution if $\gamma = \gamma_0$.

Since $f(D)$ is compact and $\eta > 0$ this scheme is necessarily finite.

5 The SIT algorithm for $\mathcal{D}(\mathcal{C})$ optimization

We now specialize the above approach to the most important classes of $\mathcal{D}(\mathcal{C})$-optimization problems, namely dc and dm optimization problems.

For the sake of convenience let us rewrite problem (P) in the form :

$$\min\{f(x)|\ g_1(x) - g_2(x) \leq 0,\ x \in [a, b]\}, \quad f, g_1, g_2 \in \mathcal{C}, \tag{P}$$

Here $\mathcal{C}$ is either the set of convex functions (when (P) is a dc optimization problem), or the set of increasing functions (when (P) is a dm optimization problem).

As was shown in Sect. 4, Procedure (*, γ) for problem (P) is an adaptive BB algorithm for solving problem (Q_γ), with deletion criterion "$\beta(M) > \varepsilon$" and stopping criterion "$g(x^k) > \varepsilon$".

Incorporating Procedure (*, γ) into the SIT Scheme (Successive Incumbent Transcending Scheme) we can state the following SIT Algorithm for (P).

Let γ_0 be any real number such that $\gamma > \max\{f(x)|\ x \in [a, b]\}$.

SIT Algorithm for (P)

Select tolerances $\varepsilon > 0, \eta > 0$.

Step 0 Let $\mathcal{P}_1 = \{M_1\}$, $M_1 = [a, b]$, $\mathcal{R}_1 = \emptyset$. Let $\gamma = \gamma_0$. Set $k = 1$.

Step 1 For each box (hyperrectangle) $M \in \mathcal{P}_k$:

- Reduce M, i.e. find a box $[p, q] \subset M$ as small as possible satisfying $\min\{g(x)|\ f(x) \leq \gamma, x \in [p, q]\} = \min\{g(x)|\ f(x) \leq \gamma, x \in M\}$ and set $M \leftarrow [p, q]$.
- Compute a lower bound $\beta(M)$ for $g(x)$ over the feasible solutions in $[p, q]$. ($\beta(M)$ must be such that: $\beta(M) < +\infty \Rightarrow \{x \in M|\ f(x) \leq \gamma\} \neq \emptyset$, see (6)) Delete every M such that $\beta(M) > \varepsilon$.

Step 2 Let $\mathcal{P}'_k$ be the collection of boxes that results from $\mathcal{P}_k$ after completion of Step 1. Let $\mathcal{R}'_k = \mathcal{R}_k \cup \mathcal{P}'_k$.

Step 3 If $\mathcal{R}'_k = \emptyset$ then *terminate*: if $\gamma = \gamma_0$ the problem (P) is ε-infeasible; otherwise, the nonisolated feasible solution $\bar{x}$ such that $\gamma = f(\bar{x}) - \eta$ is an essential (ε, η)-optimal solution of (P).

Step 4 If $\mathcal{R}'_k \neq \emptyset$, let $M_k \in \operatorname{argmax}\{\alpha(M)|\ M \in \mathcal{R}'_k\},\ \beta_k = \beta(M_k)$. Determine $x^k \in M_k$ and $y^k \in M_k$ such that

$$f(x^k) \leq \gamma, \quad g(y^k) - \beta(M_k) = o(\|x^k - y^k\|). \tag{18}$$

If $g(x^k) < \varepsilon$, go to Step 5. If $g(x^k) \geq \varepsilon$, go to Step 6.

Step 5 x^k is a nonisolated feasible solution satisfying $f(x^k) \leq \gamma$. Define $\bar{x} = x^k,\ \gamma = f(\bar{x}) - \eta$ and go to Step 6.

Step 6 Divide M_k into two subboxes by the adaptive bisection, i.e. bisect M_k via (v^k, j_k), where $j_k \in \operatorname{argmax}\{|y^k_j - x^k_j| : j = 1, \ldots, n\}$, $v^k = \frac{1}{2}(x^k + y^k)$. Let $\mathcal{P}_{k+1}$ be the collection of these two subboxes of M_k, $\mathcal{R}_{k+1} = \mathcal{R}'_k \setminus \{M_k\}$. Increment k, and return to Step 1.

Proposition 9 *The SIT algorithm terminates after finitely many steps, yielding an essential (ε, η)-optimal solution or an evidence that the problem has no ε-feasible solution.*

Proof Follows from the discussion in Sect. 4. □

5.1 Discussion

Several operations mentioned in Steps 1 and 4 of the above algorithm should be discussed in more detail.

5.1.1 DC optimization (g_1, g_2, f are convex functions)

Valid reduction Let $M = [p, q]$ be a box. We wish to determine a box $\operatorname{red}M = [p', q'] \subset M$ such that

$$\{x \in [p', q']|\ f(x) \leq \gamma,\ g_1(x) - g_2(x) \leq \varepsilon\} = \{x \in M|\ f(x) \leq \gamma,\ g_1(x) - g_2(x) \leq \varepsilon\}. \tag{19}$$

Select a convex function $\bar{g}(x)$ underestimating $g(x) = g_1(x) - g_2(x)$ on M and tight at a point $y \in M$, i.e. such that $\bar{g}(y) = g(y)$. For example $\bar{g}(x) = g_1(x) - g_2(y)$, where y is a corner of the hyperrectangle $[p, q]$ maximizing $g_2(x)$ over $[p, q]$). Then p', q' are determined by solving, for $i = 1, \ldots, n$, the convex programs:

$$p'_i = \min\{x_i|\ x \in [p, q],\ f(x) \leq \gamma,\ \bar{g}(x) \leq \varepsilon\},$$
$$q'_i = \max\{x_i|\ x \in [p, q],\ f(x) \leq \gamma,\ \bar{g}(x) \leq \varepsilon\}.$$

Bounding and Condition (18) With $\bar{g}(x)$ being a convex underestimator of $g(x) = g_1(x) - g_2(x)$ over $M = [p, q]$, tight at a point $y \in \text{red}M$ define $\beta(M) = \min\{\bar{g}(x)|\ f(x) \le \gamma, x \in \text{red}M\}$. So for each k there is a point $y^k \in M_k$ such that $\bar{g}(y^k) = g(y^k)$. Also define $x^k \in \text{argmin}\{\bar{g}(x)|\ f(x) \le \gamma, x \in [p^k, q^k]\}$, so that $\bar{g}(x^k) = \beta(M_k)$. It can easily be checked that condition (18) is satisfied.

Alternative Method: Define

$$x^M \in \text{argmin}\{g_1(x)|\ f(x) \le \gamma, x \in \text{red}M\}$$
$$y^M \in \text{argmax}\{g_2(x)|\ x \in \text{red}M\}$$
$$\beta(M) = g_1(x^M) - g_2(y^M).$$

The points $x^k = x^{M_k}$, $y^k = y^{M_k}$ satisfy (18).

Special case: When f, g are polynomials or signomials, one can define at each iteration k a convex program (e.g. a SDP)

$$\max\{\ell_k(x, u)|\ f(x) \le \gamma,\ (x, u) \in C_k\}, \tag{L$_k$}$$

where $u \in \mathbb{R}^{m_k}$, C_k is a compact set in $M_k \times \mathbb{R}^{m_k}$, and $\ell_k(x, u)$ is a linear function such that the problem $\max\{g(x)|\ f(x) \le \gamma, x \in M_k\}$ is equivalent to

$$\max\{\ell_k(x, u)|\ f(x) \le \gamma,\ (x, u) \in C_k, r_i(x, u) \ge 0\ (i \in I_k)\}, \tag{NL$_k$}$$

where $r_i(x, u) \ge 0,\ i \in I_k$, are nonconvex constraints [3,4]. Furthermore, (L$_k$) can be selected so that there exists $y^k \in M_k$ satisfying

$$(y^k, u) \in C_k \Rightarrow r_i(y^k, u) \ge 0\ \ (i \in I_k). \tag{20}$$

(see e.g. [7], Proposition 8.12). In that case, setting

$$\alpha(M_k) = \max\{\ell_k(x, u)|\ f(x) \le \gamma,\ (x, u) \in C_k\}$$

and taking an optimal solution (x^k, u^k) of (L$_k$) one has a couple $x^k,\ y^k$ satisfying (20) and

$$x^k \in M_k,\ f(x^k) \le \gamma,\ (x^k, u^k) \in C_k,\ \ell_k(x^k, u^k) = \alpha(M_k). \tag{21}$$

If $x^k = y^k$ then from (20) and $(x^k, u^k) \in C_k$ it follows that $r_i(x^k, u) \ge 0\ (i \in I_k)$, so (x^k, u^k) solves (NL$_k$) and hence x^k solves the subproblem $\min\{g(x)|\ f(x) \le \gamma, x \in M_k\}$, i.e. $g(x^k) = \ell_k(x^k, u^k) = \alpha(M_k)$. It can then easily be shown that $g(y^k) - \beta(M_k) \to 0$ as $x^k - y^k \to 0$, so that x^k, y^k satisfy condition (18). This suggests that by passing to the robust approach, both RLT and Lasserre's methods could be refined so as to allow adaptive subdivision and thereby improve convergence.

5.1.2 *DM optimization (g_1, g_2, f are increasing functions)*

Valid reduction Let $M = [p, q]$ be the box to be reduced. Since g_1, g_2, f are increasing functions, we have (see [11], Proposition 2.11; also [14], Lemma 16):

Proposition 10 (i) *If $f(p) > \gamma$, or $g_1(q) - g_2(p) < \varepsilon$ then* red$M = \emptyset$.
(ii) *If $f(p) \le \gamma$ and $g_1(q) - g_2(p) \ge \varepsilon$ then* red$M = [p', q']$ *with*

$$p' = q - \sum_{i=1}^{n} \alpha_i (q_i - p_i) e^i, \quad q' = p' + \sum_{i=1}^{n} \alpha_i (q_i - p'_i) e^i \tag{22}$$

where, for $i = 1, \ldots, n$,

$$\alpha_i = \sup\{\alpha|\ 0 < \alpha \leq 1,\ g_1(q - \alpha(q_i - p_i)e^i) \geq g_2(p) + \varepsilon \tag{23}$$

$$\alpha_i = \sup\{\alpha|\ 0 < \alpha \leq 1,\ g_2(p' + \alpha(q_i - p'_i)e^i) \leq g_1(q) - \varepsilon,\\ f(p' + \alpha(q_i - p'_i)e^i) \leq \gamma\}. \tag{24}$$

Proof See [11]. □

Bounding and Condition (18) We can assume that M has been reduced, so that, according to Proposition 10, $f(p) \leq \gamma$ and $g_1(q) - g_2(p) \geq \varepsilon$. One method for bounding and achieving condition (18) is then to take a polyblock $P_k \supset \{x \in [p, q]|\ f(x) \leq \gamma\}$ with vertex set V_k and define x^k, y^k such that

$$x^k \in \text{argmin}\{g_1(x)|\ x \in [p, q]\},\ \text{i.e. } x^k = p, \tag{25}$$

$$y^k \in \text{argmax}\{g_2(x)|\ x \in V_k\}, \tag{26}$$

$$\beta(M_k) = g_1(p) - g_2(y^k). \tag{27}$$

Clearly, with x^k, y^k and $\beta(M)$ so defined, if $\|y^k - x^k\| \to 0$ then $\beta(M_k) \to g(x^*)$, with $x^* = \lim x^k = \lim y^k$, hence (18) is satisfied. In particular, if P_k is taken to be $M_k = [p, q]$ then $V_k = \{q\}$, so $\beta(M_k) = g_1(p) - g_2(q)$ and the adaptive subdivision simply reduces to the standard bisection.

5.2 Cases of equality constraints

The SIT Algorithm has been developed for problems (P), with only constraints of inequality type: $g_i(x) \leq 0, i = 1, \ldots, m$. In the case when there are equality constraints such as

$$h_j(x) = 0, \quad j = 1, \ldots, s,$$

one can use linear equalities to eliminate certain variables, so without loss of generality it can always be assumed that all equality constraints are nonlinear. Since an exact solution to a nonlinear system of equations cannot be expected to be computable in finitely many steps, one should be content with replacing every given equality constraint $h_j(x) = 0$ by an approximate one:

$$-\delta \leq h_j(x) \leq \delta, \quad j = 1, \ldots, s,$$

where $\delta > 0$ is the tolerance. A mixed system with both inequality and equality constraints can thus be replaced with an approximate system involving only inequality constraints to which the above approach can be applied.

Another case worth mentioning is when, aside from the constraints $g_i(x) \leq 0$, $i = 1, \ldots, m$, there is an additional convex constraint of the form $x \in C$, where C is a closed convex subset of $\mathbb{R}^n$. The extension of the above approach to this case is straightforward, by observing that the pair of problems to be considered in Proposition 7 now becomes

$$\min\{f(x)|\ g(x) \leq \varepsilon,\ x \in C \cap [a, b]\}$$
$$\min\{g(x)|\ f(x) \leq \gamma,\ x \in C \cap [a, b]\}.$$

6 Illustrative examples

To illustrate the practicality of the approach we present some numerical examples. All the computations were made on a PC Pentium IV 2.53 GHz, RAM 256Mb DDR. Some more examples can be found in our earlier papers [12,13,15].

Example 1 Minimize x_1 subject to

$$(x_1 - 5)^2 + 2(x_2 - 5)^2 + (x_3 - 5)^2 \leq 18$$

$$(x_1 + 7 - 2x_2)^2 + 4(2x_1 + x_2 - 11)^2 + 5(x_3 - 5)^2 \geq 100$$

This problem has one isolated feasible point (1, 3, 5) which is also the optimal solution.

An essential (0.01, 0.01)-optimal solution is (3.747692, 7.171420, 2.362317) with objective function value 3.747692. It was found after 33.703 sec., 15736 iterations.

Example 2 ($n = 4$) Minimize $(3 + x_1x_3)(x_1x_2x_3x_4 + 2x_1x_3 + 2)^{2/3}$ subject to

$$\begin{aligned}
&- 3(2x_1x_2 + 3x_1x_2x_4)(2x_1x_3 + 4x_1x_4 - x_2)\\
&\quad - (x_1x_3 + 3x_1x_2x_4)(4x_3x_4 + 4x_1x_3x_4 + x_1x_3 - 4x_1x_2x_4)^{1/3}\\
&\quad + 3(x_4 + 3x_1x_3x_4)(3x_1x_2x_3 + 3x_1x_4 + 2x_3x_4 - 3x_1x_2x_4)^{1/4} \leq -309.219315\\
&- 2(3x_3 + 3x_1x_2x_3)(x_1x_2x_3 + 4x_2x_4 - x_3x_4)^2\\
&\quad + (3x_1x_2x_3)(3x_3 + 2x_1x_2x_3 + 3x_4)^4 - (x_2x_3x_4 + x_1x_3x_4)(4x_1 - 1)^{3/4}\\
&\quad - 3(3x_3x_4 + 2x_1x_3x_4)(x_1x_2x_3x_4 + x_3x_4 - 4x_1x_2x_3 - 2x_1)^4 \leq -78243.910551\\
&- 3(4x_1x_3x_4)(2x_4 + 2x_1x_2 - x_2 - x_3)^2\\
&\quad + 2(x_1x_2x_4 + 3x_1x_3x_4)(x_1x_2 + 2x_2x_3 + 4x_2 - x_2x_3x_4 - x_1x_3)^4 \leq 9618\\
&\quad 0 \leq x_i \leq 5 \; i = 1, 2, 3, 4.
\end{aligned}$$

This problem would be difficult to solve by the RLT or Lasserre's method, since a very large number of variables would have to be introduced.

For $\eta = 0.01$ the essential η-optimal solution

$$x^{\text{essopt}} = (4.994594,\ 0.020149,\ 0.045424,\ 4.928073)$$

with essential η-optimal value 5.906278 was found by the SIT Algorithm at iteration 667, and confirmed as such at iteration 866. The computation required 6.343 s. and went through 53 cycles of incumbent transcending, with intermediate results for the first ten and last ten cycles as given in Table 1.

By cycle we mean a sequence of iterations required for transcending a given incumbent; $\bar{x}$ is the new incumbent found at the end of the cycle, and Iter indicates the iteration where $\bar{x}$ is found.

Example 3 ($n = 5$) Minimize

$$4(x_1^2x_3 + 2x_1^2x_2x_3^2x_5 + 2x_1^2x_2x_3)(5x_1^2x_3x_4^2x_5 + 3x_2)^{3/5} + 3(2x_4^2x_5^2)(4x_1^2x_4 + 4x_2x_5)^{5/3}$$

subject to

$$\begin{aligned}
&-2(2x_1x_5 + 5x_1^2x_2x_4^2x_5)(3x_1x_4x_5^2 + 5 + 4x_3x_5^2)^{1/2} \leq -7684.470329\\
&2(2x_1x_2^2x_3x_4^2)(2x_1x_2x_3x_4^2 + 2x_2x_4^2x_5 - x_1^2x_5^2)^{3/2} \leq 1286590.314422\\
&0 \leq x_i \leq 5 \qquad i = 1, 2, 3, 4, 5.
\end{aligned}$$

Table 1 Computational results for Example 2

Cycle	$\bar{x}$	$f(\bar{x})$	Iter
1	(4.999998, 1.271816, 3.759639, 1.249999)	368.410194	29
2	(3.857994, 1.271816, 3.759639, 1.249999)	250.293433	30
3	(4.999998, 1.263772, 1.284919, 2.499999)	101.124566	32
4	(4.994678, 1.257941, 1.279076, 2.494081)	100.113179	33
5	(4.496902, 0.733341, 1.968319, 0.690469)	99.111902	35
6	(4.999999, 0.640076, 0.663701, 3.124999)	38.898361	53
7	(4.996012, 0.635929, 0.659562, 3.120769)	38.509352	54
8	(4.999999, 0.640076, 0.663701, 2.81250)	37.385499	56
9	(4.999999, 0.336423, 0.663701, 2.812500)	32.706640	57
10	(4.999999, 0.336423, 0.364193, 2.812500)	18.249278	58
11	(4.705440, 0.336423, 0.364193, 2.812500)	17.328990	59
.	.	.	.
.	.	.	.
.	.	.	.
44	(4.977291, 0.075587, 0.065822, 3.804033)	6.530734	307
45	(4.970758, 0.069047, 0.061690, 3.773551)	6.400772	309
46	(4.999462, 0.027206, 0.061940, 3.553025)	6.336764	375
47	(4.976755, 0.046627, 0.058138, 4.089212)	6.273397	434
48	(4.986031, 0.044224, 0.055855, 4.086719)	6.210662	435
49	(4.999631, 0.021968, 0.054891, 4.018098)	6.148556	481
50	(4.971402, 0.059281, 0.049940, 4.670158)	6.087070	531
51	(4.977333, 0.037443, 0.049073, 4.711604)	6.026199	569
52	(4.994001, 0.026386, 0.047363, 4.738176)	5.965937	609
53	(4.994594, 0.020149, 0.045424, 4.928073)	5.906278	667

With $\eta = 0.01$ the essential η-optimal solution

$$\bar{x} = (4.987557,\ 4.984973,\ 0.143546,\ 1.172267,\ 0.958926)$$

with objective function value 28766.057367 was found at iteration 45119 and confirmed as such at iteration 101836. The computation took 514.422 s. and the maximal number of nodes of the branch and bound tree active at one iteration was 6853.

7 Conclusion

Global optimization problems with nonconvex constraint sets are difficult in two major respects: (1) a feasible solution may be very hard to determine, so most current methods can only find an approximate optimal solution, which quite often is infeasible and is not guaranteed to approach the exact optimal solution in an acceptable sense; (2) the optimal solution may be an isolated feasible solution, in which case a correct solution cannot be computed by a finite procedure and implemented practically.

To cope with these difficulties the robust approach proposed in this paper consists basically in finding a nonisolated feasible solution and improving it step by step. The resulting algorithm works its way to the best nonisolated optimal solution through a number of cycles of incumbent transcending. A major advantage of it, apart from stability, is that when prematurely stopped it may still provide a good nonisolated feasible solution, in contrast to current methods which are almost useless in that case.

Although the approach has been developed for $\mathcal{D}(\mathcal{C})$-optimization, it is very general because, as we have argued, virtually every nonconvex global optimization problem of genuine interest can be cast as a $\mathcal{D}(\mathcal{C})$-optimization problem. Also, as can easily be verified, this approach can be applied to any problem (P) where the objective function $f(x)$ is such that for any real number γ the level set $\{x \in [a, b]| \ f(x) \leq \gamma\}$ has no isolated point.

References

1. Audet, C., Hansen, P., Jaumard, B., Savard, G.: A branch and cut algorithm for nonconvex quadratically constrained quadratic programming. Math. Progr. Ser. A **87**, 131–152 (2000)
2. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches. 3rd edn. Springer, Berlin (1996)
3. Lasserre, J.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**, 796–817 (2001)
4. Sherali, H., Adams, W.: A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems. Kluwer, Dordrecht (1999)
5. Tikhonov, A.N.: On a reciprocity principle. Sov. Math. Dokl. **22**, 100–103 (1980)
6. Tuy, H.: Convex programs with an additional reverse convex constraint. J. Optim. Theory Appl. **52**, 463–486 (1987)
7. Tuy, H.: Convex Analysis and Global Optimization. Kluwer, Dordrecht (1978)
8. Tuy, H.: Monotonic optimization: problems and solution approaches. SIAM J. Optim. **11**(2), 464–494 (2000)
9. Tuy, H., Hoai Phuong, N.T.: A unified monotonic approach to generalized linear fractional programming. J. Glob. Optim. **23**, 1–31 (2002)
10. Tuy, H., Thach, P.T., Konno, H.: Optimization of polynomial fractional functions. J. Glob. Optim. **29**, 19–44 (2004)
11. Tuy, H., Al-Khayyal, F., Thach, P.T.: Monotonic optimization: branch and cut methods. In: Audet, C., Hansen, P., Savard, G. Essays and Surveys on Global Optimization, Springer, Berlin, pp. 39–78 (2005)
12. Tuy, H.: Robust solution of nonconvex global optimization problems. J. Glob. Optim. **32**, 307–323 (2005)
13. Tuy, H.: Polynomial optimization: a robust approach. Pac. J. Optim. **1**, 357–374 (2005)
14. Tuy, H., Minoux, M., Hoai-Phuong, N.T.: Discrete monotonic optimization with application to a discrete location problem. SIAM J. Optim. **17**, 78–97 (2006)
15. Tuy, H., Hoai-Phuong, N.T.: A robust algorithm for quadratic optimization under quadratic constraints. J. Glob. Optim. **37**, 557–569 (2007)