Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER Applied Mathematics and Computation 169 (2005) 247–277

APPLIED MATHEMATICS AND COMPUTATION

www.elsevier.com/locate/amc

Optimal centers in branch-and-prune algorithms for univariate global optimization

D.G. Sotiropoulos *, T.N. Grapsa

University of Patras, Department of Mathematics, Division of Computational Mathematics and Informatics, GR-265 04 Rio, Patras, Greece

Abstract

We present an interval branch-and-prune algorithm for computing verified enclosures for the global minimum and all global minimizers of univariate functions subject to bound constraints. The algorithm works within the branch-and-bound framework and uses first order information of the objective function. In this context, we investigate valuable properties of the optimal center of a mean value form and prove optimality. We also establish an inclusion function selection criterion between natural interval extension and an optimal mean value form for the bounding process. Based on optimal centers, we introduce linear (inner and outer) pruning steps that are responsible for the branching process. The proposed algorithm incorporates the above techniques in order to accelerate the search process. Our algorithm has been implemented and tested on a test set and compared with three other methods. The method suggested shows a significant improvement on previous methods for the numerical examples solved.

Keywords: Global optimization; Branch-and-prune; Optimal centers; Optimal mean value form; Pruning steps

* Corresponding author.
E-mail address: dgs@math.upatras.gr (D.G. Sotiropoulos).

0096-3003/$ - see front matter
doi:10.1016/j.amc.2004.10.050

1. Introduction

Interval branch-and-bound methods for global optimization provide guaranteed and reliable solutions for global optimization problems of the form

$$\min_{x \in X} f(x),$$

where $f : \mathscr{D} \subseteq \mathbb{R} \to \mathbb{R}$ is a C^1 function, $\mathscr{D}$ is the closure of a nonempty bounded open subset of $\mathbb{R}$, and $X \subseteq \mathscr{D}$ is a search interval representing bound constraints for x. Our aim is to find the global minimum f^* and the set $X^* = \{x^* \in X : f(x^*) = f^*\}$ of all global minimizers of f. Since minimizing $f(x)$ is equivalent to maximizing $-f(x)$, the above definition includes the search for global maxima, too.

Based on the branch-and-bound principle [14], these methods subdivide the search interval into subintervals and use bounds for the objective function to exclude from consideration subintervals where a global minimizer cannot lie. Interval arithmetic [17,1] provides rigorous bounds for the range $f_{\mathrm{rg}}(X)$ of f over X. Finding sharper bounds with a reasonable amount of computation is a central problem. When the objective function is continuously differentiable, interval enclosures of the derivative along with a mean value form is often used to improve the enclosure of the function range [17,5,23]. Moreover, derivative bounds may also be used for testing the monotonicity of the function in order to discard subregions where the function is strictly monotone [10,14].

Branch-and-prune methods employ successive decomposition of the initial problem into smaller disjoint subproblems that are solved recursively until a termination criterion is fulfilled and the optimal solution is found. Branching is accomplished either by subdividing or, implicitly, by pruning the search region. In [12], Hansen et al. describe an algorithm that utilizes high order derivatives to compute the so-called cord-slopes which are used by several tests to eliminate parts of the search area, under the assumption that the objective function has only finitely many local minima. In contrast, pruning techniques that utilize only first order information by means of interval derivatives are given in [27,28,7,29]. For nonsmooth functions, a pruning technique, given by Ratz [26], based on interval slopes [15], can be applied.

In this paper we present a branch-and-prune algorithm that uses first-order information by means of an interval derivative evaluation. The algorithm heavily depends on the existence of optimal centers in mean value forms. We determine optimal centers in a straightforward manner and prove, as an alternative to Baumann [4], that the mean value form becomes optimal when supplied with optimal centers (Theorem 3). We also prove optimality for the mean value form with respect to the width (Theorem 4) and establish a selection criterion between natural interval extension and optimal mean value form for the bounding process (Proposition 9). Optimal centers are also used during the pruning process. In Section 3, we introduce a new accelerating device that composes *inner* and *outer pruning steps* to eliminate parts of the search interval where global minimum points do not exist. The proposed algorithm incorporates the above ideas to accelerate the search process. It utilizes first order information fourfold: (i) to check monotonicity, (ii) to calculate optimal centers, (iii) to apply optimal mean value form for bounding the range of the function, and (iv) to prune the search interval using the current upper bound of the global minimum.

For a thorough study of the underlining theory see [23,10,14,19]. For the rest of the paper, however, some notations and notions are needed. Let $\mathbb{IR} = \{[\underline{x}, \bar{x}] | \underline{x} \leqslant \bar{x},\ \underline{x}, \bar{x} \in \mathbb{R}\}$ be the set of compact intervals. The lower bound $\underline{x}$ is also referred to as inf X and the upper bound $\bar{x}$ as sup X. We define the midpoint as $m(X) = (\underline{x} + \bar{x})/2$ and the radius of X as $r(X) = (\bar{x} - \underline{x})/2$. We call a function $F : \mathbb{IR} \to \mathbb{IR}$ an *inclusion function* of $f : \mathbb{R} \to \mathbb{R}$ if $f_{\mathrm{rg}}(X) \subseteq F(X)$ for any $X \in \mathbb{IR}$. The inclusion function of the derivative of f is denoted by F'. Inclusion functions can be produced in a number of ways such as natural extension, mean value forms and Taylor expansion. Each of these forms has slightly different properties and convergence order. For a more thorough discussion on these issues see [22].

2. Optimal mean value forms

The mean value form is a particular type of inclusion functions, apart from the natural interval extension F, that provides bounds of the range f_{rg} of the objective function f. The definition of the mean value form F_{MVF} of f is as follows (see [17,19]):

Definition 1. Let $f : \mathscr{D} \to \mathbb{R}$ be a C^1 function, $X \subseteq \mathscr{D} \subseteq \mathbb{R}$ and F' an interval inclusion of f' on X. The interval inclusion $F_{\mathrm{MVF}} : \mathbb{IR} \times \mathbb{R} \to \mathbb{IR}$ of f on X, defined by

$$F_{\mathrm{MVF}}(X, c) = f(c) + F'(X) \cdot (X - c), \tag{1}$$

is the mean value form of f on X with center c.

The mean value form depends on the derivative f' of f, thus it can be used only if the function is differentiable. Now, the estimation of the range $f_{\mathrm{rg}}(X)$ has been reduced to the computation of an enclosure $F'(X)$ of f', which can be obtained easily by applying automatic differentiation in connection with interval arithmetic [9]. In [5,20] it is shown that the mean value form offers a second order approximation to the range of f, i.e., $w(F_{\mathrm{MVF}}(X, c)) - w(f_{\mathrm{rg}}(X)) = \mathscr{O}(w(X)^2)$. Compared with the natural interval extension F of f, the mean value form F_{MVF} assures tighter enclosures when the interval X is narrow. On the contrary, F_{MVF} may drastically overestimate the range of f when the width of X is large. For this reason, it is a common practice to use the intersection $F_{\mathrm{MVF}}(X,c) \cap F(X)$ for a better estimation of the range. A more extensive discussion on these issues is given in Section 2.1.

The selection of the center c in (1) is a crucial task that permits a more general definition of the mean value form and leads to different inclusions of $f_{\mathrm{rg}}(X)$. A common approach is to set $c = m(X)$. In this case, the mean value form is expressed as the sum of $f(c)$ and a symmetric interval, where $f(c)$ is the center of the interval enclosure. In [4], Baumann defines (using simple asymmetric expressions) the novel notion of the optimal center and proves that, when this particular point is the center of a mean value form, then the form is optimal. We describe these notions in the sequel. Terminology is borrowed from [4].

Definition 2. The mean value form $F_{\mathrm{MVF}}(X,c) = f(c) + F'(X) \cdot (X - c)$ is called

(i) optimal with respect to the lower bound if

$$\inf F_{\mathrm{MVF}}(X,c) \leqslant \inf F_{\mathrm{MVF}}(X,c^-) \text{ for any } c \in X, \text{ and}$$

(ii) optimal with respect to the upper bound if

$$\sup F_{\mathrm{MVF}}(X,c^+) \leqslant \sup F_{\mathrm{MVF}}(X,c) \text{ for any } c \in X.$$

The point c^- which yields the greatest lower bound and the point c^+ which yields the lowest upper bound among all centers are called optimal centers.

In [5,20] it is shown that, if F' is an isotone inclusion of f' and c is the midpoint of X, then the mean value form is inclusion isotone, i.e., $X \subseteq Y$ implies $F_{\mathrm{MVF}}(X,c) \subseteq F_{\mathrm{MVF}}(Y,c)$. Baumann's optimal centers do not preserve isotonicity; instead, a weaker property called *one-sided isotonicity* holds for optimal mean value forms (for the definition, see [4]). One-sided (lower or upper bound) isotonicity requires that at least one bound is improved while inclusion isotonicity requires that both lower and upper bounds are improved simultaneously. In global minimization algorithms, lower bound isotonicity suffices.

In [4, Theorem 1], Baumann proves that optimal centers c^- and c^+ lead to optimal mean value forms with respect to lower and upper bound, respectively. Here, we study the opposite direction: we seek for these particular centers that make a mean value form optimal. Our approach is straightforward: we compute the center for which the lower (upper) bound of the mean value form (1) attains its maximum (minimum). We finally see that these points are the optimal centers of Baumann.

Based on the definition of the infimum and supremum of the mean value form (1), if we set $X = [\underline{x}, \bar{x}]$ and $F'(X) = [\underline{d}, \bar{d}]$ and apply the interval multiplication rules, the lower bound of the mean value form (1) is

$$\inf F_{\mathrm{MVF}}(X,c) = f(c) + \begin{cases} \underline{d}(\bar{x}-c) & \text{if } \bar{d} \leqslant 0, \\ \bar{d}\ (\underline{x}-c) & \text{if } \underline{d} \geqslant 0, \\ \min(\underline{d}(\bar{x}-c), \bar{d}(\underline{x}-c)) & \text{otherwise,} \end{cases} \tag{2a–c}$$

while the upper bound is

$$\sup F_{\mathrm{MVF}}(X,c) = f(c) + \begin{cases} \underline{d}(\underline{x}-c) & \text{if } \bar{d} \leqslant 0, \\ \bar{d}(\bar{x}-c) & \text{if } \underline{d} \geqslant 0, \\ \max(\underline{d}(\underline{x}-c), \bar{d}(\bar{x}-c)) & \text{otherwise.} \end{cases} \tag{3a–c}$$

Theorem 3. *Let $f : \mathscr{D} \to \mathbb{R}$ be a C^1 function with $\mathscr{D} \subseteq \mathbb{R}$, $X = [\underline{x}, \bar{x}] \subseteq \mathscr{D}$, and $F'(X) = [\underline{d}, \bar{d}]$ be an enclosure of the derivative of f over X.*

(*i*) *The lower bound of the mean value form attains its maximum at the center*

$$c^- = \begin{cases} \bar{x} & \textit{if } \bar{d} \leqslant 0, \\ \underline{x} & \textit{if } \underline{d} \geqslant 0, \\ m(X) - r(X) \cdot \dfrac{\bar{d}+\underline{d}}{\bar{d}-\underline{d}} & \textit{otherwise} \end{cases} \tag{4a–c}$$

and for all $c \in X$,

$$-w(F'(X)) \cdot |c^- - c| \leqslant \inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-) \leqslant 0. \tag{5}$$

(*ii*) *The upper bound of the mean value form attains its minimum at the center*

$$c^+ = \begin{cases} \underline{x} & \textit{if } \bar{d} \leqslant 0, \\ \bar{x} & \textit{if } \underline{d} \geqslant 0, \\ m(X) + r(X) \cdot \dfrac{\bar{d}+\underline{d}}{\bar{d}-\underline{d}} & \textit{otherwise} \end{cases} \tag{6a–c}$$

and for all $c \in X$,

$$0 \leqslant \sup F_{\mathrm{MVF}}(X,c) - \sup F_{\mathrm{MVF}}(X,c^+) \leqslant w(F'(X)) \cdot |c^+ - c|. \tag{7}$$

Proof. (i) Our aim is to determine the value of $c \in X$ such that the lower bound of the mean value form, given by (2), to be maximized. For a given interval X, let h be a function of c with $h(c) = \inf F_{\mathrm{MVF}}(X,c)$. There are three possible cases: If $\bar{d} \leqslant 0$, then $h(c) = f(c) + \underline{d}(\bar{x} - c)$ is monotonically nondecreasing in X since $h'(c) \geqslant 0$ for all $c \in X$. Thus, h attains its maximum value at the point $c = \bar{x}$. By similar arguments, when $\underline{d} \geqslant 0$, the function h is monotonically nonincreasing in X and is maximized at the point $c = \underline{x}$.

In the last case (the most interesting one), when $\underline{d} < 0 < \bar{d}$ then $h(c) = f(c) + \mathrm{L}(c)$, where $\mathrm{L}(c) = \min(\ell_1(c), \ell_2(c))$, $\ell_1(c) = \underline{d}(\bar{x} - c)$ and $\ell_2(c) = \bar{d}(\underline{x} - c)$.

It is easy to show that L is a piecewise linear and concave function on the closed interval X and, hence, any local maximum is also a global one. The graphical representations of ℓ_1, ℓ_2 and L are depicted in Fig. 1. The maximizer c^- of L is the abscissa of the intersection point of the linear functions ℓ_1 and ℓ_2. Since $\underline{x} = m(X) - r(X)$ and $\bar{x} = m(X) + r(X)$, after some trivial manipulations we obtain that $c^- = m(X) - r(X) \cdot (\bar{d} + \underline{d})/(\bar{d} - \underline{d})$. It remains still to show that c^- is also the global maximizer of h. Function h is continuous on X but not differentiable at the cusp point c^- since the left-hand and the right-hand derivatives at this point are not equal. Moreover, h is strictly monotone increasing on the open interval $(\underline{x}, c^-)$ and strictly monotone decreasing on $(c^-, \bar{x})$. Therefore, h attains its maximum at the point c^-, too.

To prove the inequality (5), we have

$$\begin{aligned} &\inf F_{\mathrm{MVF}}(X, c) - \inf F_{\mathrm{MVF}}(X, c^-) \\ &\quad = f(c) - f(c^-) + \inf(F'(X)(X - c)) - \inf(F'(X)(X - c^-)). \end{aligned} \tag{8}$$

By applying the mean value theorem, we have $f(c) - f(c^-) \in F'(X)(c - c^-)$ and hence,

$$f(c) - f(c^-) \leqslant \sup(F'(X)\ (c - c^-)) = \max(\underline{d}\ (c - c^-),\ \bar{d}\ (c - c^-)) \tag{9}$$

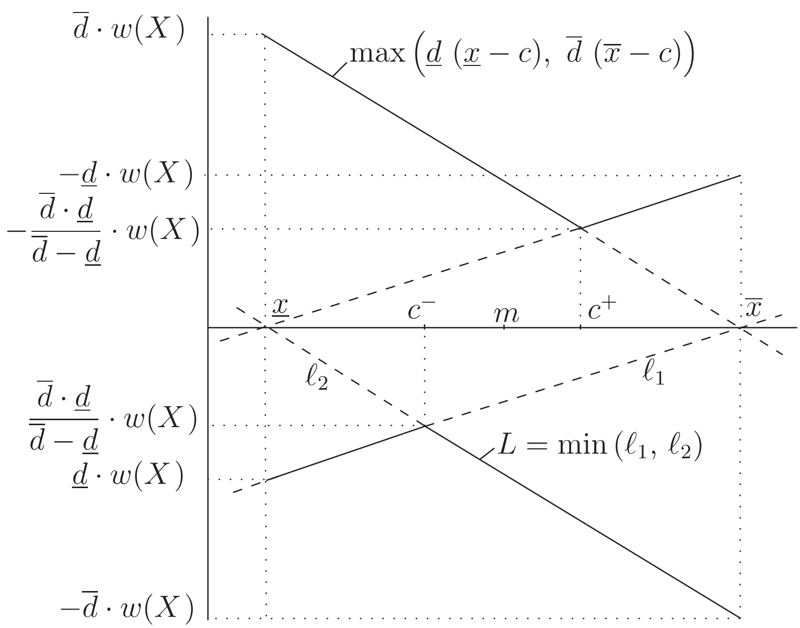

Fig. 1. Graphical presentation of functions ℓ_1, ℓ_2, and L when $\underline{d} < 0 < \bar{d}$.

and

$$f(c) - f(c^-) \geqslant \inf(F'(X)(c - c^-)) = \min(\underline{d}\ (c - c^-),\ \bar{d}(c - c^-)). \quad (10)$$

Using inequality (9) along with (8) we obtain

$$\begin{aligned}\inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-) \leqslant{}& \max(\underline{d}(c - c^-), \bar{d}(c - c^-)) \\ &+ \min(\underline{d}(\bar{x} - c), \bar{d}(\underline{x} - c)) \\ &- \min(\underline{d}(\bar{x} - c^-), \bar{d}(\underline{x} - c^-)).\end{aligned} \quad (11)$$

Taking into account the formulas that hold $\forall\ \alpha_1, \alpha_2 \in \mathbb{R}$

$$\min(\alpha_1, \alpha_2) = (\alpha_1 + \alpha_2 - |\alpha_1 - \alpha_2|)/2 \quad \text{and} \quad (12)$$

$$\max(\alpha_1, \alpha_2) = (\alpha_1 + \alpha_2 + |\alpha_1 - \alpha_2|)/2, \quad (13)$$

inequality (11) becomes

$$\inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-) \leqslant (|\alpha| - |\beta| + |\gamma|)/2, \quad (14)$$

where $\alpha = (\bar{d} - \underline{d})(c^- - c)$, $\beta = \underline{d}(\bar{x} - c) - \bar{d}(\underline{x} - c)$ and $\gamma = \underline{d}(\bar{x} - c^-) - \bar{d}(\underline{x} - c^-)$. But $\alpha + \beta = \gamma = 0$ (from the definition of c^-). Therefore, $|\alpha| = |\beta|$ and thus

$$\inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-) \leqslant 0 \quad \text{for all } c \in X.$$

To complete the proof, Eq. (8), combined with inequality (10) and formula (12), results in

$$(-|\alpha| - |\beta| + |\gamma|)/2 \leqslant \inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-). \quad (15)$$

Since $|\alpha| = |\beta|$ and $|\gamma| = 0$, $-|\alpha| \leqslant \inf F_{\mathrm{MVF}}(X,c) - \inf F_{\mathrm{MVF}}(X,c^-)$ and the proof of inequality (5) is complete.

(ii) The optimal center c^+ can be similarly obtained by minimizing the upper bound of the mean value form given in (3). The proof of inequality (7) is similar as before. □

In view of (4) and (6) it is obvious that the centers c^- and c^+ are always in the interval X and they are symmetric with respect to the midpoint $m(X)$ (see Fig. 1). Moreover, their relative position within X depends only on the sign of the sum $\bar{d} + \underline{d}$. Specifically,

$$\begin{aligned} &c^- < m(X) < c^+, && \text{if } \bar{d} + \underline{d} > 0, \\ &c^+ < m(X) < c^-, && \text{if } \bar{d} + \underline{d} < 0 \text{ and} \\ &c^- \equiv m(X) \equiv c^+, && \text{if } \bar{d} + \underline{d} = 0. \end{aligned}$$

Baumann observes that the width of the mean value form (1) is minimal for all centers that are between c^- and c^+ [19, p. 75]. The next theorem states that this observation is true and reveals another significant property of the optimal mean value form.

Theorem 4. *Let $f : \mathscr{D} \to \mathbb{R}$ be a C^1 function, $X = [\underline{x}, \bar{x}] \in \mathbb{IR}$ and $F'(X) = [\underline{d}, \bar{d}]$ be an enclosure of the derivative of f over X. Then, the mean value form*

$$F_{\mathrm{MVF}}(X, c) = f(c) + F'(X) \cdot (X - c), \quad \textit{for } c \in X$$

has minimal width for all centers which are in the interval with endpoints c^- and c^+.

Proof. We utilize a simplified version of Rall's midpoint-radius formula [21, p. 222] that computes the width $w(A \cdot B)$ of the product of two intervals $A, B \in \mathbb{IR}$ [16,24]:

$$w(A \cdot B) = \max\{|m(A)|w(B) + w(A)|m(B)|, |m(A)|w(B) + w(A)w(B)/2, w(A)|m(B)| + w(A)w(B)/2\}. \tag{16}$$

For $A = F'(X)$ and $B = X - c$, relation (16) gives

$$w(F_{\mathrm{MVF}}) = \max\{|\bar{d} + \underline{d}| \cdot w(X)/2 + (\bar{d} - \underline{d}) \cdot |m(X) - c|, |\bar{d} + \underline{d}| \cdot w(X)/2 + (\bar{d} - \underline{d}) \cdot w(X)/2, (\bar{d} - \underline{d}) \cdot |m(X) - c| + (\bar{d} - \underline{d}) \cdot w(X)/2\}. \tag{17}$$

Our task is to minimize $\varphi(c) = w(F_{\mathrm{MVF}})$ with respect to $c \in X$. This corresponds to the simple minimax problem

$$\min_{c \in X} \varphi(c),$$

$$\varphi(c) = \max(\varphi_1(c), \varphi_2(c), \varphi_3(c)),$$

where $\varphi_1(c) = \alpha|m-c| + \beta_1$, $\varphi_2(c) = \beta_2$, $\varphi_3(c) = \alpha|m-c| + \beta_3$, $\alpha = \bar{d} - \underline{d}$, $\beta_1 = |\bar{d} + \underline{d}|w/2$, $\beta_2 = (|\bar{d} + \underline{d}| + \bar{d} - \underline{d})w/2$ and $\beta_3 = (\bar{d} - \underline{d})w/2$. See that $\alpha \geqslant 0$, $0 \leqslant \beta_1 \leqslant \beta_3 \leqslant \beta_2$, and $\beta_2 = \beta_1 + \beta_3$.

The graphs of φ_1, φ_2, and φ_3 are depicted in Fig. 2 when $\bar{d} + \underline{d} > 0$. Function φ represented by the polygonal line ABCD is piecewise linear and convex, and attains its minimum value for all c between B and C. By solving the equation $\varphi_2(c) = \varphi_3(c)$, it can be easily verified that the abscissas of B and C are equal to c^- and c^+, respectively. When $\bar{d} + \underline{d} < 0$, the abscissas of B and C are equal to c^+ and c^-, respectively. Finally, when $\bar{d} + \underline{d} = 0$, then $\beta_3 = \beta_2$ and the (unique) global minimum point of $\varphi(c)$ is the midpoint of the interval. □

Let us now derive the bounds of $F_{\mathrm{MVF}}(X, c^-)$ in an explicit form. The following corollaries can be easily obtained by substituting c^- and c^+ in relations (2) and (3).

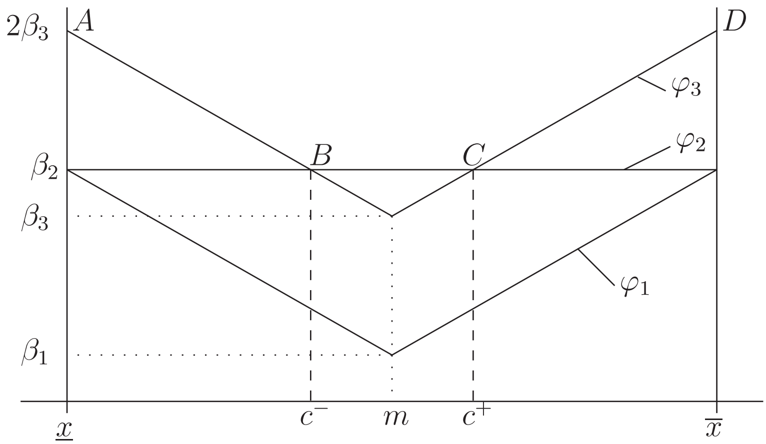

Fig. 2. The graphs of φ_1, φ_2, and φ_3 when $\bar{d} + \underline{d} > 0$.

Corollary 5. *The lower bound of the mean value form* (1) *with center* c^- *is*

$$\inf F_{\mathrm{MVF}}(X, c^-) = \begin{cases} f(\bar{x}) & \textit{if } \bar{d} \leqslant 0, \\ f(\underline{x}) & \textit{if } \underline{d} \geqslant 0, \\ f(c^-) + w(X) \cdot \dfrac{\bar{d} \cdot \underline{d}}{\bar{d} - \underline{d}} & \textit{otherwise} \end{cases} \tag{18a–c}$$

and the upper bound is

$$\sup F_{\mathrm{MVF}}(X, c^-) = \begin{cases} f(\bar{x}) - w(X) \cdot \underline{d} & \textit{if } \bar{d} \leqslant 0, \\ f(\underline{x}) + w(X) \cdot \bar{d} & \textit{if } \underline{d} \geqslant 0, \\ f(c^-) + w(X) \cdot \dfrac{\max(\bar{d}^2, \underline{d}^2)}{\bar{d} - \underline{d}} & \textit{otherwise.} \end{cases} \tag{19a–c}$$

Corollary 6. *The lower bound of the mean value form* (1) *with center* c^+ *is*

$$\inf F_{\mathrm{MVF}}(X, c^+) = \begin{cases} f(\underline{x}) + w(X) \cdot \underline{d} & \textit{if } \bar{d} \leqslant 0, \\ f(\bar{x}) - w(X) \cdot \bar{d} & \textit{if } \underline{d} \geqslant 0, \\ f(c^+) + w(X) \cdot \dfrac{\min(\bar{d}^2, \underline{d}^2)}{\bar{d} - \underline{d}} & \textit{otherwise} \end{cases} \tag{20a–c}$$

and the upper bound is

$$\sup F_{\mathrm{MVF}}(X, c^+) = \begin{cases} f(\underline{x}) & \textit{if } \bar{d} \leqslant 0, \\ f(\bar{x}) & \textit{if } \underline{d} \geqslant 0, \\ f(c^+) - w(X) \cdot \dfrac{\bar{d} \cdot \underline{d}}{\bar{d} - \underline{d}} & \textit{otherwise.} \end{cases} \tag{21a–c}$$

When conditions (18a) or (18b) hold, function f is monotonic and hence, inf $F_{\mathrm{MVF}}(X,c^-)$ is the exact lower bound. The same holds for sup $F_{\mathrm{MVF}}(X,c^+)$ when (21a) or (21b) is the case. On the contrary, when the midpoint $m(X)$ is used as the center in the mean value form, then no exact bounds can be achieved (see (2a) and (2b)). When derivative bounds guarantee that f is monotonic, the *optimal bicentered form* [19, p. 59]

$$F_{\mathrm{BF}}(X, c^-, c^+) = F_{\mathrm{MVF}}(X, c^-) \cap F_{\mathrm{MVF}}(X, c^+)$$

returns the range of the function without overestimation; the corresponding bounds are given by (18) and (21). The optimal bicentered form can also be used to get a better enclosure for the range of a function in the general case when $\underline{d} < 0 < \bar{d}$. Nevertheless, the price to pay for this improvement is one more function evaluation at point c^+.

We further investigate the properties of the optimal centers in connection with the mean value form. The next proposition supplies upper and lower bounds in a more definite shape when the relative position of c^- within X is taken into account.

Proposition 7. *The lower and upper bound of the mean value form* (1) *with center* c^- *are expressed by the following relations*:

$$\inf F_{\mathrm{MVF}}(X, c^-) = f(c^-) + \begin{cases} \bar{d}\ (\underline{x} - c^-) & \textit{if } \bar{d} + \underline{d} < 0, \\ \underline{d}(\bar{x} - c^-) & \textit{if } \bar{d} + \underline{d} > 0, \\ (-1) \cdot r(X) \cdot r(F'(X)) & \textit{otherwise} \end{cases} \tag{22a–c}$$

and

$$\sup F_{\mathrm{MVF}}(X, c^-) = f(c^-) + \begin{cases} \underline{d}\ (\underline{x} - c^-) & \textit{if } \bar{d} + \underline{d} < 0, \\ \bar{d}\ (\bar{x} - c^-) & \textit{if } \bar{d} + \underline{d} > 0, \\ r(X) \cdot r(F'(X)) & \textit{otherwise}. \end{cases} \tag{23a–c}$$

Proof. We reformulate the interval $F_{\mathrm{MVF}}(X,c^-)$ in the midpoint-radius coordinate system. Keeping in mind the definition of c^- in (4) and using (18) and (19), it is easy to verify that the radius of $F_{\mathrm{MVF}}(X,c^-)$ is

$$r(F_{\mathrm{MVF}}(X, c^-)) = \begin{cases} r(X) \cdot (-\underline{d}) & \text{if } \bar{d} + \underline{d} < 0, \\ r(X) \cdot \bar{d} & \text{if } \bar{d} + \underline{d} > 0, \\ r(X) \cdot r(F'(X)) & \text{otherwise}, \end{cases} \tag{24a–c}$$

while the midpoint of $F_{\mathrm{MVF}}(X, c^-)$ is

$$m(F_{\mathrm{MVF}}(X, c^-)) = \begin{cases} f(c^-) + \underline{d} \cdot (m(X) - c^-) & \text{if } \bar{d} + \underline{d} < 0, \\ f(c^-) + \bar{d} \cdot (m(X) - c^-) & \text{if } \bar{d} + \underline{d} > 0, \\ f(c^-) & \text{otherwise.} \end{cases} \tag{25a–c}$$

Coming back to the endpoints coordinate system, relations (22) and (23) follow immediately. □

Notice that relations (22) and (23) estimate lower and upper bounds of f in a more general way than relations (18) and (19). When $\bar{d} \leqslant 0$ or $\underline{d} \geqslant 0$, relations (22) and (23) are reduced to (18) and (19), respectively. After a closer look one can see that when (22b) or (22a) is the case, the underestimation of f with respect to $f(c^-)$ is the same. In general, expression (22) reveals that there is no need for the width $w(X)$ of the interval X to tend to zero, to have a sharp lower bound.

Let us examine what happens in the simple case of a quadratic function. We next reproduce a lemma given by Baumann in [3].

Lemma 8. *Let $f : X \to \mathbb{R}$ be a quadratic function of the form $f(x) = ax^2 + b\,x + c$, $X = [\underline{x}, \bar{x}] \in \mathbb{IR}$ and $0 \in F'(X)$. Then*

$$\min_{x \in X} f(x) = f(c^-).$$

Proof. The parabola $ax^2 + b\,x + c$ attains its minimum at the point $x^* = -b/2a$. Since, $F'(X) = 2aX + b$, by applying relation (4) one can easily verify that $c^- = -b/2a$ and therefore $f(c^-) = f(x^*)$. □

Lemma 8 shows that for a quadratic function f the optimal center c^- coincides with the minimizer x^* of f. On the other hand, expression (22) shows that the range of f can be still overestimated. To eliminate overestimation, we subdivide the interval X at c^-. Then a quadratic f becomes monotone in both subintervals and Proposition 7 ensures exact lower bounds in both of them. This fact motivates for adopting a subdivision strategy that utilizes the optimal center c^- instead of the midpoint of the interval (described in Section 4.2).

2.1. Optimal mean value form vs natural interval extension

A natural question that arises when one minimizes a function f over an interval X, is to decide which inclusion function gives the best lower bound for the range of f. There are two common choices: the natural interval extension $F(X)$ of f and a mean value form $F_{\mathrm{MVF}}(X, c)$ when an interval enclosure of f' is available. Theorems 3 and 4 show that optimal mean value form

$F_{\mathrm{MVF}}(X, c^{-})$ is the best choice among all one-centered mean value forms. Composite mean value forms, like the *linear bound value form* [19, p. 60] and the *kite inclusion function* [29], with more than one center also exist. However, the price to pay is more function evaluations.

In this section, we compare natural interval extension and optimal mean value form. The next proposition suggests a selection criterion between the optimal mean value form and the natural interval extension at a given state of the computation.

Proposition 9. *Let $f : X \to \mathbb{R}$ be a C^1 function, $X = [\underline{x}, \bar{x}] \in \mathbb{IR}$ and $0 \in F'(X) = [\underline{d}, \bar{d}]$ be an enclosure of the derivative of f over X. Assuming that the bounds of the natural interval extension F(X) are available,*

$$\text{if } w(F(X)) \leqslant \lambda \cdot w(X) \text{ then } \inf F_{\mathrm{MVF}}(X, c^{-}) \leqslant \inf F(X),$$

where $\lambda = -(\bar{d} \cdot \underline{d})/(\bar{d} - \underline{d})$ is a nonnegative constant.

Proof. Since $w(F(X)) \leqslant \lambda \cdot w(X)$, $\sup F(X) \leqslant \inf F(X) + \lambda \cdot w(X)$. Taking into account that $\inf F(X) \leqslant f(c^{-}) \leqslant \sup F(X)$ along with relation (18), we conclude that $\inf F_{\mathrm{MVF}}(X, c^{-}) \leqslant \inf F(X)$. □

It is obvious that the value of λ depends on the bounds of $F'(X)$. These bounds are usually calculated with automatic differentiation. The automatic computation of $F'(X)$ requires the computation of $F(X)$, which is performed by direct interval evaluation of the inclusion function F. In branch-and-bound optimization methods the bounds of the derivatives are used essentially in the monotonicity test. However, if the monotonicity test is inconclusive then a lower bound of the function f must be determined. Proposition 9 guarantees that when the hypothesis is true, the optimal mean value form gives no better lower bound than the natural interval extension $F(X)$ and hence, an extra (and useless) function evaluation at the optimal center c^{-} is avoided.

On the other hand, if the hypothesis is false, i.e., $w(F(X)) > \lambda \cdot w(X)$, the selection of the most preferable inclusion function remains an open issue. One can easily verify that when $f(c^{-}) > \inf F(X) + \lambda \cdot w(X)$ then $\inf F(X) < \inf F_{\mathrm{MVF}}(X, c^{-})$. Unfortunately, this condition involves a function evaluation at c^{-}. Thus, the only way to obtain the best lower bound is to calculate the intersection of $F(X)$ with $F_{\mathrm{MVF}}(X, c^{-})$.

3. Linear pruning steps

In [11], Hansen and Sengupta have determined the feasible region for an inequality constrained problem by solving a set of interval linear inequalities. Inspired by their work, we developed a process that iteratively prunes the search interval, called *inner and outer pruning steps*. This pruning technique is based on first-order information (by means of an interval evaluation of the derivative) and serves as an accelerating device in the main part of our algorithm. We next present the theoretical aspects as well as algorithmic formulations for the linear pruning (inner and outer) steps. As we show, inner and outer pruning steps arise from the solution of linear interval inequalities obtained from first-order Taylor expansions.

Let $\tilde{f}$ denote the smallest already known upper bound of the global minimum f^* and let $Y \subseteq X$ be the current subinterval. The upper bound $\tilde{f}$ can be used in an attempt to prune Y. Our aim is to find an interval enclosure $\widetilde{Y}$ of the set of points $\tilde{y}$ such that $f(\tilde{y}) \leqslant \tilde{f}$. Taking into account that $f \in C^1$, we can expand f about a center $c \in Y$, i.e., $f(\tilde{y}) = f(c) + (\tilde{y} - c) \cdot f'(\xi)$, where ξ is a point between c and $\tilde{y}$. Since $c \in Y$ and $\tilde{y} \in Y$, then $\xi \in Y$. Thus,

$$f(\tilde{y}) \in f(c) + F'(Y) \cdot (\tilde{y} - c) \leqslant \tilde{f}.$$

By setting $z = \tilde{y} - c$, $f_c = f(c)$ and $D = [\underline{d}, \bar{d}] = F'(Y)$, the above interval inequality becomes

$$(f_c - \tilde{f}) + D \cdot z \leqslant 0. \tag{26}$$

The solution set Z of inequality (26) is the following (cf. [11,10]):

$$Z = \begin{cases} [-\infty,\ (\tilde{f} - f_c)/\bar{d}] \cup & \\ [(\tilde{f} - f_c)/\underline{d}, +\infty], & \text{if } \tilde{f} < f_c,\ \underline{d} < 0 < \bar{d}, \\ [-\infty,\ (\tilde{f} - f_c)/\bar{d}], & \text{if } \tilde{f} < f_c,\ \underline{d} \geqslant 0 \text{ and } \bar{d} > 0, \\ [(\tilde{f} - f_c)/\underline{d}, +\infty], & \text{if } \tilde{f} < f_c,\ \underline{d} < 0 \text{ and } \bar{d} \leqslant 0, \\ \emptyset, & \text{if } \tilde{f} < f_c,\ \underline{d} = \bar{d} = 0, \\ [-\infty, +\infty], & \text{if } \tilde{f} \geqslant f_c,\ \underline{d} \leqslant 0 \leqslant \bar{d}, \\ [-\infty,\ (\tilde{f} - f_c)/\underline{d}], & \text{if } \tilde{f} \geqslant f_c,\ \underline{d} > 0, \\ [(\tilde{f} - f_c)/\bar{d}, +\infty], & \text{if } \tilde{f} \geqslant f_c,\ \bar{d} < 0. \end{cases} \tag{27a–g}$$

$\widetilde{Y} = c + Z$ is the set of points $\tilde{y} \in Y$ that satisfy inequality (26). Since we are only interested in points $\tilde{y} \in Y$, we compute the interval enclosure $\widetilde{Y}$ as

$$\widetilde{Y} = (c + Z) \cap Y. \tag{28}$$

The last two cases of (27) are not of interest since then the function is strictly monotone in the entire subinterval and thus it cannot contain any stationary points in its interior. We next deal with the rest of the cases: If Z can be represented as the union of two intervals, say Z_1 and Z_2 (case (27a)), then the interval enclosure $\widetilde{Y}$ is composed of the intervals $\widetilde{Y}_i = (c + Z_i) \cap Y$, $i = 1,2$. In different cases, where Z is a single interval, the desired solution of inequality (26) is a single interval $\widetilde{Y}$ and as a consequence, the current subinterval Y is pruned either from the right (case (27b)) or from the left (case (27c)). Case

(27d) is a special one that occurs only when f is constant and $\underline{d} = \bar{d} = 0$ within Y. Since $\tilde{f} < f_c$, interval Y cannot contain any global minimizer and can be discarded at once.

3.1. Inner pruning step

In this subsection we algorithmically formulate the *inner pruning step*. This step utilizes derivative bounds and an extra function value at a point $c \in Y \subseteq X$. For the case of interest where $0 \in D = [\underline{d}, \bar{d}]$, we set

$$p = c + (\tilde{f} - f_c)/\bar{d} \quad \text{and} \quad q = c + (\tilde{f} - f_c)/\underline{d}.$$

Then, when $\tilde{f} < f_c$ holds, the relation (28) in connection with (27) takes the following form:

$$\widetilde{Y} = \begin{cases} [\underline{y}, p] \cup [q, \bar{y}], & \text{if } \underline{d} < 0 < \bar{d}, \\ [\underline{y}, p], & \text{if } \underline{d} \geqslant 0 \text{ and } \bar{d} > 0, \\ [q, \bar{y}], & \text{if } \underline{d} < 0 \text{ and } \bar{d} \leqslant 0, \\ \emptyset, & \text{if } \underline{d} = \bar{d} = 0. \end{cases} \tag{29a–d}$$

When $\tilde{f} \geqslant f_c$ and $\underline{d} \leqslant 0 \leqslant \bar{d}$, the interval enclosure $\widetilde{Y}$ coincides with $\widetilde{Y} = [\underline{y}, \bar{y}]$, and hence, no real pruning is possible.

In case (29a), the interval enclosure $\widetilde{Y}$ is composed of the intervals $Y_1 = [\underline{y}, p]$ and $Y_2 = [q, \bar{y}]$. Since $\tilde{f} - f_c < 0$, $p < c < q$, the point p expresses the leftmost point such that $\tilde{f} < f(p)$, while q is the rightmost point such that $\tilde{f} < f(q)$. Thus, the global minimum cannot lie in the open interval (p, q) and therefore the interval (p, q) can be discarded with guarantee (see Fig. 3). In case (29b) or (29c), the interval enclosure $\widetilde{Y}$ is the single interval Y_1 or Y_2, respectively; the gap intervals $(p, \bar{y}]$ and $[\underline{y}, q)$ can also be discarded. When (29d) is the case, then the whole interval is discarded, since $\widetilde{Y} = \emptyset$. The properties of the interval $\widetilde{Y}$ are summarized in the next theorem:

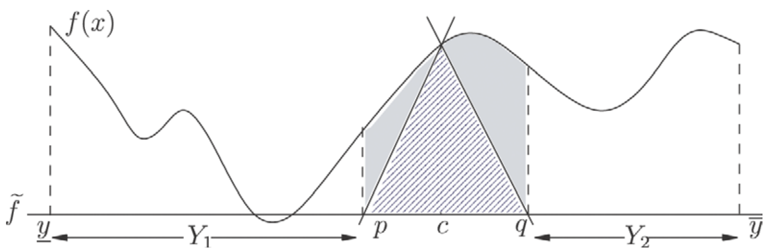

Fig. 3. Geometric interpretation of the inner pruning step when $\underline{d} < 0 < \bar{d}$.

Theorem 10. *Let* $f : \mathscr{D} \to \mathbb{R}$ *be a* C^1 *function,* $Y \in \mathbb{IR}$, $c \in Y \subseteq X \subseteq \mathscr{D} \subseteq \mathbb{R}$. *Moreover, let* $f_c = f(c)$, $0 \in D = F'(Y)$ *and* $\tilde{f} \geqslant \min_{x \in X} f(x)$. *Then, the interval* $\widetilde{Y}$ *defined by* (29) *has the following properties*:

1. $\widetilde{Y} \subseteq Y$.
2. *Every global optimizer* x^* *of* f *in* Y *satisfies* $x^* \in \widetilde{Y}$.
3. *If* $\widetilde{Y} = \emptyset$ *then there exists no global minimizer of* f *in* Y.

Algorithm 1. The inner pruning algorithm.

InnerPrune $(Y, c, f_c, D, \tilde{f}, Y_1, Y_2)$
if $\bar{d} > 0$ **then**
  $p := c + (\tilde{f} - f_c)/\bar{d}$;
  if $p \geqslant \underline{y}$ **then**
    $Y_1 := [\underline{y}, p]$;
  else
    $Y_1 := \emptyset$;
if $\underline{d} < 0$ **then**
  $q := c + (\tilde{f} - f_c)/\underline{d}$;
  if $q \leqslant \bar{y}$ **then**
    $Y_2 := [q, \bar{y}]$;
  else
    $Y_2 := \emptyset$;
return Y_1, Y_2;

Proof. Property 1 follows immediately from the definition of $\widetilde{Y}$ in (29). The proof of Property 2 is implied by the above discussion, while Property 3 is a consequence of Property 2. □

Based on cases (29a)–(29d), where a pruning is possible under the necessary condition $\tilde{f} < f_c$, we can now formulate the detailed steps of the inner pruning algorithm. Algorithm 1 takes as input the subinterval $Y = [\underline{y}, \bar{y}] \subseteq X$, $c \in Y$, $f_c = f(c)$, the derivative enclosure $D = [\underline{d}, \bar{d}]$ over Y and the current upper bound $\tilde{f}$, and returns the pruned (possibly empty) subset $\widetilde{Y} = Y_1 \cup Y_2$ of Y.

In Fig. 3 we illustrate the inner pruning step by drawing two lines from the point $(c, f(c))$ and intersecting them with the $\tilde{f}$ level. The first line with the largest (positive) slope $\bar{d}$ of f in Y intersects the $\tilde{f}$ level at point p, while the line with the smallest slope $\underline{d}$ intersects the $\tilde{f}$ level at point q. Notice that, in a geometrical sense, the inner pruning step is equivalent to one step of the extended interval Newton's method [17,1,10] applied to equation $f(y) - \tilde{f} = 0$, $y \in Y$.

3.2. Outer pruning step

Outer pruning step utilizes the known information in an attempt to contract the bounds of the current subinterval $Y \subseteq X$. Let us assume, at the moment, that the function values $f(\underline{y})$ and $f(\bar{y})$ at the two boundary points are available and $0 \in D = F'(Y)$. Expanding f about the endpoints $\underline{y}$ and $\bar{y}$, we obtain two interval inequalities similar to (26):

$$(f(\underline{y}) - \tilde{f}) + D \cdot z_{\mathrm{L}} \leqslant 0 \quad \text{and} \quad (f(\bar{y}) - \tilde{f}) + D \cdot z_{\mathrm{R}} \leqslant 0, \tag{30}$$

where $z_{\mathrm{L}} = \tilde{y}_{\mathrm{L}} - \underline{y}$ and $z_{\mathrm{R}} = \tilde{y}_{\mathrm{R}} - \bar{y}$. In a recent paper, Casado et al. [7] have made the key observation that it is possible to prune the current interval without additional cost if we know only a lower bound for the value of f at the boundary points, while the exact function values themselves are not necessary. We exploit this observation in developing the outer pruning step. If $\underline{f_{\mathrm{L}}}$ and $\underline{f_{\mathrm{R}}}$ are the lower bounds of $f(\underline{y})$ and $f(\bar{y})$, respectively, then the inequalities (30) are equivalent to

$$(\underline{f_{\mathrm{L}}} - \tilde{f}) + D \cdot z_{\mathrm{L}} \leqslant 0 \quad \text{and} \quad (\underline{f_{\mathrm{R}}} - \tilde{f}) + D \cdot z_{\mathrm{R}} \leqslant 0. \tag{31}$$

Our aim is to find the set $\widetilde{Y}$ of points $\tilde{y}$ satisfying both inequalities (31). Thus, we request $\widetilde{Y} = \widetilde{Y}_{\mathrm{L}} \cap \widetilde{Y}_{\mathrm{R}}$ where, according to (27) and (28),

$$\begin{aligned} \widetilde{Y}_{\mathrm{L}} &= ((\underline{y} + Z_{\mathrm{L}}) \ \cap \ Y) \\ &= ([-\infty, \underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\bar{d}] \cup [\underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}, +\infty]) \cap [\underline{y}, \bar{y}] \\ &= ([-\infty, \underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\bar{d}] \cap [\underline{y}, \bar{y}]) \cup ([\underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}, +\infty] \cap [\underline{y}, \bar{y}]) \\ &= \emptyset \cup [\underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}, \bar{y}] = [\underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}, \bar{y}] \end{aligned} \tag{32}$$

and

$$\widetilde{Y}_{\mathrm{R}} = ((\bar{y} + Z_{\mathrm{R}}) \cap Y) = [\underline{y}, \bar{y} + (\tilde{f} - \underline{f_{\mathrm{R}}})/\bar{d}]. \tag{33}$$

The properties of the interval $\widetilde{Y}$ are summarized in the next theorem:

Theorem 11. *Let $f : \mathscr{D} \to \mathbb{R}$ be a C^1 function, $Y = [\underline{y}, \bar{y}] \subseteq X \subseteq \mathbb{R}$ and $0 \in F'(Y) = [\underline{d}, \bar{d}]$. Let also $\underline{f_{\mathrm{L}}}$ and $\underline{f_{\mathrm{R}}}$ be the lower bound of $f(\underline{y})$ and $f(\bar{y})$, respectively, and $\tilde{f} \geqslant \min_{x \in X} f(x)$ be the current upper bound such that $\tilde{f} \leqslant \min\{\underline{f_{\mathrm{L}}}, \underline{f_{\mathrm{R}}}\}$. Then, the interval*

$$\widetilde{Y} = [r, s] = [\underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}, \ \bar{y} + (\tilde{f} - \underline{f_{\mathrm{R}}})/\bar{d}] \tag{34}$$

has the following properties:

1. $\widetilde{Y} \subseteq Y$.
2. *Every global optimizer* x^* *of* f *in* Y *satisfies* $x^* \in \widetilde{Y}$.
3. *If* $\widetilde{Y} = \emptyset$, *then there exists no global optimizer of* f *in* Y.

Proof. Property 1 immediately follows from the definition of the interval $\widetilde{Y}$ in (34). For the proof of Property 2, assume that there exists an $x^* \in \mathscr{X} = Y \cap [\underline{y}, r)$ such that $f(x^*) = f^*$. Then, there will be a $d_* \in [\underline{d}, \bar{d}]$ satisfying $f(x^*) = f(\underline{y}) + d_*(x^* - \underline{y})$. Thus,

$$\begin{aligned} f(x^*) &= f(\underline{y}) + d_* \cdot (x^* - \underline{y}) \geq f(\underline{y}) + \underline{d} \cdot (x^* - \underline{y}) > \underline{f_{\mathrm{L}}} + \underline{d} \cdot (x^* - \underline{y}) \\ &> \underline{f_{\mathrm{L}}} + \underline{d} \cdot (r - \underline{y}) = \underline{f_{\mathrm{L}}} + (\tilde{f} - \underline{f_{\mathrm{L}}}) = \tilde{f}, \end{aligned}$$

which contradicts $\tilde{f} \geqslant \min_{x \in X} f(x)$. Therefore $x^* \notin \mathscr{X}$. The case $x^* \notin \mathscr{X} = Y \cap (s, \bar{y}]$ can be processed similarly. Finally, Property 3 follows from Properties 1 and 2. $\square$

We present now the algorithmic formulation of the outer pruning step described above. Algorithm 2 takes as input the subinterval $Y = [\underline{y}, \bar{y}] \subseteq X$, the derivative enclosure $D = [\underline{d}, \bar{d}]$ over Y, the current upper bound $\tilde{f}$ and the lower bounds $\underline{f_{\mathrm{L}}}$ (left) and $\underline{f_{\mathrm{R}}}$ (right) of $f(\underline{y})$ and $f(\bar{y})$, respectively, and returns the outwardly pruned (possibly empty) interval Y.

Algorithm 2. The outer pruning algorithm.

OuterPrune $(Y, \underline{f_{\mathrm{L}}}, \underline{f_{\mathrm{R}}}, D, \tilde{f})$
if $\tilde{f} < \underline{f_{\mathrm{R}}}$ **then** {interval Y can be pruned from the right}
  if $\bar{d} > 0$ **then**
    $s := \bar{y} + (\tilde{f} - \underline{f_{\mathrm{R}}})/\bar{d}$;
    if $s \geqslant \bar{y}$ **then**
      $Y := [\underline{y}, s]$;
    else
      $Y := \emptyset$;
if $\tilde{f} < \underline{f_{\mathrm{L}}}$ **then** {interval Y can be pruned from the left}
  if $\underline{d} < 0$ **then**
    $r := \underline{y} + (\tilde{f} - \underline{f_{\mathrm{L}}})/\underline{d}$;
    if $r \leqslant \underline{y}$ **then**
      $Y := [r, \bar{y}]$;
    else
      $Y := \emptyset$;
return Y;

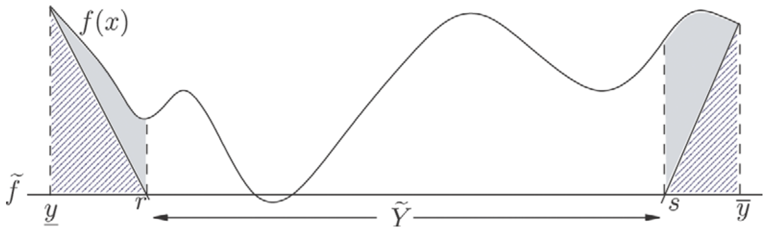

Fig. 4. Geometric interpretation of the outer pruning step.

In Fig. 4 the geometric interpretation of the outer pruning step is illustrated by using sharp lower bounds for $f(\underline{y})$ and $f(\bar{y})$. The point s is determined as the intersection of the line $h(y) = f(\bar{y}) + \bar{d} \cdot (y - \bar{y})$ with the current upper bound $\tilde{f}$ level for the global minimum. Similarly, the point r is given by the intersection of the line $g(y) = f(\underline{y}) + \underline{d} \cdot (y - \underline{y})$ with the $\tilde{f}$.

It is obvious that, given an upper bound $\tilde{f}$, the sharper the value of $\underline{f_L}$ and $\underline{f_R}$ is, the more contracted the current search interval is. The sharpest possible value of $\underline{f_L}$ and $\underline{f_R}$ is $f(\underline{y})$ and $f(\bar{y})$, respectively. However, this choice would cost two extra function evaluations. We avoid any extra computational effort by utilizing the known information. We compute the function values only for the starting interval X. After the application of Algorithm 2 at the starting interval X as $\mathsf{OuterPrune}(X, f(\underline{x}), f(\bar{x}), F'(X), \tilde{f})$, the value of $\tilde{f}$ acts as a lower bound, say $\hat{f}$, for the value of f at the endpoints of the outwardly pruned interval $Y \subseteq X$ (see Fig. 4).

3.3. The pruning steps algorithm

Algorithm 1 can be considerably enhanced when combined with Algorithm 2. The combined algorithm (Algorithm 3) aims at discarding portions of the search space in which the value of the objective function is always greater than the best known upper bound $\tilde{f}$ and thus to accelerate the optimization process. We next give a detailed description of the proposed method.

Algorithm 3 takes as input the subinterval $Y = [\underline{y}, \bar{y}]$, the common lower bound $\hat{f}$ for the value of f at the endpoints of Y, the center $c \in Y$, $f_c = f(c)$, the derivative bounds $D = [\underline{d}, \bar{d}]$ and the current upper bound $\tilde{f}$, and returns at most two subintervals Y_1 and Y_2 as well as the value of $\hat{f}$ for the generated subintervals.

Algorithm 3. The pruning algorithm.

$\mathsf{Prune}(Y, \hat{f}, c, f_c, D, \tilde{f}, Y_1, Y_2)$

1: $Y_1 := \emptyset$; $Y_2 := \emptyset$; {initialize subintervals Y_1 and Y_2}

2: **if** $\tilde{f} < f_c$ **then**

3: InnerPrune $(Y, c, f_c, D, \tilde{f}, Y_1, Y_2)$; {an inner pruning is possible}
4: **if** $Y_1 \neq \emptyset$ **then**
5: OuterPrune $(Y_1, \hat{f}, \tilde{f}, D, \tilde{f})$; {possible outer pruning in Y_1 from the left}
6: **if** $Y_2 \neq \emptyset$ **then**
7: OuterPrune $(Y_2, \tilde{f}, \hat{f}, D, \tilde{f})$; {possible outer pruning in Y_2 from the right}
8: $\hat{f} := \tilde{f}$; {set the lower bound for the created subinterval(s)}
9: **else**
10: $Y_1 := [\underline{y}, c]$; $Y_2 := [c, \bar{y}]$; {subdivide the interval at point c}
11: OuterPrune $(Y_1, \hat{f}, f_c, D, f_c)$; {outer pruning from the left for Y_1}
12: OuterPrune $(Y_2, f_c, \hat{f}, D, f_c)$; {outer pruning from the right for Y_2}
13: $\hat{f} := f_c$; {set the lower bound for subintervals Y_1 and Y_2}
14: **return** $Y_1, Y_2, \hat{f}$;

Steps 3–8 handle the case where an inner pruning is possible when the condition $\tilde{f} < f_c$ holds. After the application of the inner pruning step, an outer pruning step may be performed at the produced subintervals Y_1 and Y_2 when $\tilde{f} < \hat{f}$. This can be done by applying Algorithm 2 to each nonempty subinterval, taking into account that: (i) the function value of f at the right endpoint of Y_1 and the left endpoint of Y_2 is bounded below from $\tilde{f}$, and (ii) the function value of f at the left endpoint of Y_1 and right endpoint of Y_2 inherit the lower bound $\hat{f}$ of their ancestor Y (see Fig. 5). After the application of the outer pruning steps the endpoints of Y_1 and Y_2 are bounded below by $\tilde{f}$ (Step 8). A geometrical interpretation of Steps 3–8 is illustrated in Fig. 5 where inner and outer pruning steps are combined. Notice that no splitting is necessary since the branching process follows from the inner pruning step.

When inner pruning is not possible (Steps 10–13), Algorithm 3 forms two subintervals, Y_1 and Y_2, by subdividing Y at the point c and performs an outer pruning step at the left side of Y_1 and the right side of Y_2 when $\tilde{f} < \hat{f}$ (see Fig. 6). By this way, a sharp lower bound for the function value at the common

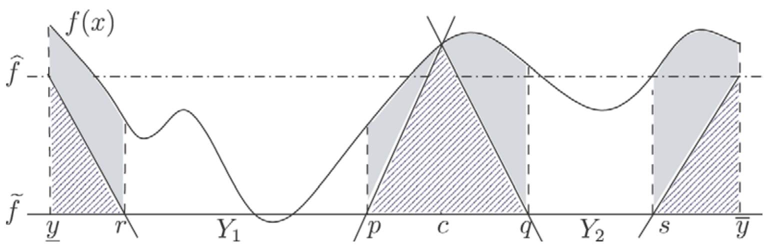

Fig. 5. Geometric interpretation of the pruning steps in the case when $\tilde{f} < f_c$.

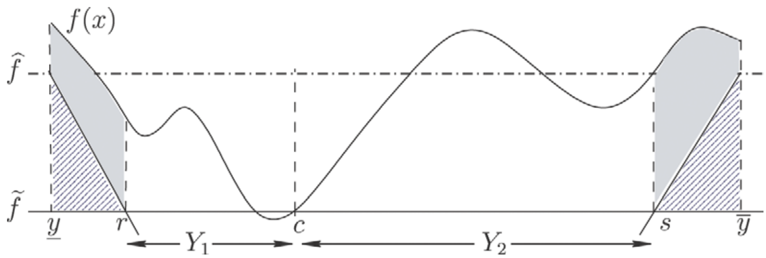

Fig. 6. Geometric interpretation of the pruning steps in the case when $\tilde{f} \geqslant f_c$.

endpoint of Y_1 and Y_2 is obtained. It must be pointed out that it is crucial to subdivide at the point c where the function value f_c is known; subdividing at any different point would require an extra function evaluation.

Algorithm 3 acts as a new accelerating device that utilizes already known information: the interval derivative $F'(Y)$ and a function value $f(c)$ used in the mean value form to obtain a lower bound for f over Y. The following theorem summarizes the properties of Algorithm 3.

Theorem 12. *Let $f : \mathscr{D} \to \mathbb{R}$ be a C^1 function, $c \in Y \subseteq X \subseteq \mathbb{R}$. Set also $f_c = f(c)$, $D = F'(Y)$ and $\hat{f} > \tilde{f} \geqslant \min_{x\in X} f(x)$. Then Algorithm 3 applied as*

$$\mathsf{Prune}\,(Y, \hat{f}, c, f_c, D, \tilde{f}, Y_1, Y_2)$$

has the following properties:

1. $Y_1 \cup Y_2 \subseteq Y$.
2. *Every global optimizer x^* of f in Y satisfies $x^* \in Y_1 \cup Y_2$.*
3. *If $Y_1 \cup Y_2 = \emptyset$, then there exists no global optimizer of f in Y.*

Proof . It follows from the definition of Y_1 and Y_2 and Theorems 10 and 11. $\square$

4. The branch-and-prune algorithm

The proposed algorithm is based on the branch-and-bound principle. Within the branch and bound framework it is useful to conceptualize the search process in terms of a tree, where each node is associated with an interval. The root of the tree is the initial search region X while each node is a subinterval of X having at most two descendants. Generally speaking, the search tree is expanded incrementally by iterating the following steps: (i) The initial search interval is subdivided into smaller subintervals, (ii) The objective function (and possibly its derivatives) is bounded over the subintervals, and (iii) Subintervals that definitely cannot contain a global minimizer are pruned (using the calculated bounds). We discuss these issues in the sequel.

4.1. Bounding the objective function

It is a common practice for bounding the objective function f over an interval X to compute the intersection of the natural extension $F(X)$ with the mean value form $F_{\mathrm{MVF}}(X,c)$ of f, where the center c is the midpoint of X. We have already seen in Section 2 that when the optimal center c^- is used, the corresponding mean value form $F_{\mathrm{MVF}}(X,c^-)$ is optimal: it supplies the maximum lower bound (Theorem 3) and has minimal width (Theorem 4). Thus, in our algorithm we calculate the intersection $F_{\mathrm{MVF}}(X,c^-)\cap F(X)$ to obtain the best lower bound for the range of f.

Regardless of the selection of the center c, the use of a mean value form requires an extra function evaluation at the point c. We have shown in Section 2.1 that this extra cost can be avoided if a certain condition holds: when $w(F(X))\leqslant\lambda\cdot w(X)$, the natural interval extension $F(X)$ suffices to obtain the best lower bound and there is no need for a mean value form to be applied (see Proposition 9).

4.2. Accelerating and pruning techniques

The efficiency of a branch-and-bound scheme lies in its ability to enumerate most of the branches implicitly. This, in turn, depends on the bounding efficiency and the pruning tests. In first-order interval methods three criteria are commonly used to ensure that a subinterval Y contains no global minimizer. Then, the corresponding node is pruned and the computation moves to another branch of the tree. We next briefly describe the interval techniques that accelerate the search process:

Function range test. An interval Y is pruned when the lower bound $\inf F(Y)$ is greater than the current upper bound $\tilde{f}$. When the range test fails to prune Y, Y is stored in a list with candidate intervals for further investigation.

Cut-off test. When $\tilde{f}$ is improved, the function range test is applied for all candidate intervals in the list. Obviously, the better the improvement of $\tilde{f}$ is, the more effective the cut-off test is.

Monotonicity test. In some cases it determines whether the objective function f is strictly monotone in an entire interval Y. If $0\notin F'(Y)$ then Y is deleted.

The above set of accelerating devices can be augmented by adding the *pruning steps* described in the previous section. As already described, Algorithm 3 is responsible for the branching process when incorporated in a branch-and-bound scheme. When all previous tests fail, Algorithm 3 not only generates the offsprings of the parent node but also it discards parts of the interval where the global minimum does not lie. In this way the search space is reduced and the whole process is further accelerated.

Notice, however, that Algorithm 3 increases the computational effort since it requires the function value $f(c)$ at some point c. This extra cost can be avoided if one exploits information gained by the bounding process. Recall that we have to calculate the function value at the optimal center c^- when we use the optimal mean value form for bounding the range of f. Since $f(c^-)$ is then known, we supply c^- (and $f(c^-)$) to Algorithm 3. In different case, we select as c the midpoint of the interval, adding one function evaluation at the total effort. In any case, a function evaluation at some point c aids us with the improvement of the upper bound $\tilde{f}$.

Algorithm 4. The selection of center c.

SelectCenter$(Y, D, F(Y), \epsilon, c)$
1: $\lambda := -\bar{d} \cdot \underline{d}/w(D)$; {compute the constant λ as in Proposition 9}
2: **if** $w(F(Y)) \leqslant \lambda \cdot w(Y)$ **then**
3: $\quad c := m(Y)$; {we select the midpoint as center}
4: **else**
5: $\quad c :=$ OptimalCenter(Y, D); {we select the optimal center c^-}
6: $\quad$ **if** $w(Y) > \epsilon$ **then**
7: $\quad\quad$ **if** $(c - \underline{y}) < \epsilon$ **then**
8: $\quad\quad\quad c := \underline{y} + \epsilon$; {$c$ tends to the left endpoint}
9: $\quad\quad$ **if** $(\bar{y} - c) < \epsilon$ **then**
10: $\quad\quad\quad c := \bar{y} - \epsilon$; {$c$ tends to the right endpoint}
11: **return** c;

Algorithm 4 formulates the way we select the point c. Most of the steps of the algorithm are self explanatory, except from Steps 6–10 that are responsible for adjusting the optimal center c^- when it tends to the left or the right endpoint of the interval, i.e., when $\underline{d} \to 0^-$ or $\bar{d} \to 0^+$, respectively (recall the definition of c^- in relation (4)). If a subdivision at c^- was allowed (see Algorithm 3, Step 10), the algorithm would make no progress. Instead, we adjust c^- in such a way that an interval with width ϵ (the predefined tolerance) is formed when a subdivision at this point occurs.

4.3. Termination criterion

To guarantee finite termination of the algorithm we have to specify a stopping criterion. An interval Y is accepted as a solution interval and is added in the result list $\mathscr{L}^*$ if one of the following criteria holds: either the relative diameter of Y or the relative diameter of $F(Y)$ is less than a specified tolerance ϵ. In the first case, the interval Y is small enough and thus, rounding errors may prevent further refinements of the bounds of $F(Y)$, while in the latter case the function is flat on Y. The definition of the relative diameter can be found in [9], while for alternative termination criteria see [23, p. 81].

4.4. List organization

The efficiency of a branch-and-prune method heavily depends on the way the search tree is traversed. In our scheme, a search is performed according to the best-first strategy, where the interval with the smallest value of the inclusion function is selected. The candidate subintervals Y_i are stored in a *working list* $\mathscr{L}$. The elements of $\mathscr{L}$ are sorted in nondecreasing order with respect to their lower bound $\inf F(Y_i)$ and in decreasing order with respect to their age (cf. [9]). The rationale behind this rule is that since we aim at finding the global minimum, we must concentrate on the most promising interval, the one with the lowest lower bound. An alternative selection strategy, called RejectIndex, has been proposed and extensively studied in [6,8].

4.5. Description of the algorithm

We give now a detailed algorithmic formulation of the proposed global optimization method. All ideas described in the previous sections are incorporated in Algorithm 5. The algorithm takes as input the objective function f, the initial search interval X and the tolerance ϵ, and returns an interval F^* containing the global minimum f^*, along with the result list $\mathscr{L}^*$ of intervals containing all the global minimizers.

It should be emphasized that interval arithmetic is used to evaluate $f(c)$ in order to bound all rounding errors. Moreover, special care has been taken over correct rounding while computing the pruning points in Algorithms 1 and 2 (for a thorough discussion on these issues see [25]).

Initially, the working list $\mathscr{L}$ and the result list $\mathscr{L}^*$ are empty. After the evaluation of $F'(X)$, Algorithm 4 is called to supply a center c and the upper bound $\tilde{f}$ is initialized as the minimum of $\sup F(\underline{x})$, $\sup F(\overline{x})$ and $\sup F(c)$.

Algorithm 5. The branch-and-prune algorithm.

GlobalOptimize $(f, X, \epsilon, F^*, \mathscr{L}^*)$
1: $\mathscr{L} = \{\}; \quad \mathscr{L}^* = \{\};$
2: SelectCenter $(X, F'(X), F(X), \epsilon, c);$
3: $\tilde{f} := \min\{\sup F(\underline{x}), \sup F(\overline{x}), \sup F(c)\};$
4: **if** $\tilde{f} \geqslant \sup F(\underline{x})$ **then**
5: $\mathscr{L}^* := \mathscr{L}^* \uplus ([\underline{x}, \underline{x}], \inf F(\underline{x}));$

6: **if** $\tilde{f} \geqslant \sup F(\bar{x})$ **then**
7: $\mathscr{L}^* := \mathscr{L}^* \uplus ([\bar{x}, \bar{x}], \inf F(\bar{x}));$
8: $\mathsf{OuterPrune}\,(X, \inf F(\underline{x}), \inf F(\bar{x}), F'(X), \tilde{f});$
9: $\hat{f} := \tilde{f};$
10: $F_X := (F(c) + F'(X)(X - c)) \cap F(X);$
11: $\mathscr{L} := \mathscr{L} \uplus (X, \inf F_X, F'(X), c, F(c), \hat{f});$
12: **while** $\mathscr{L} \neq \{\}$ **do**
13: $\quad (Y, \inf F_Y, F'(Y), c, F(c), \hat{f}) := \mathsf{PopHead}\ (\mathscr{L});$
14 $\quad \mathsf{Prune}\,(Y, c, \inf F(c), F'(Y), \tilde{f}, \hat{f}, Y_1, Y_2);$
15: $\quad$ **for** $i := 1$ to 2 **do**
16: $\qquad$ **if** $Y_i = \emptyset$ **then next** i;
17: $\qquad$ **if** $0 \notin F'(Y_i)$ **then next** i;
18: $\qquad \mathsf{SelectCenter}\,(Y_i, F'(Y_i), F(Y_i), \epsilon, c);$
19: $\qquad$ **if** $\sup F(c) < \tilde{f}$ **then**
20: $\qquad\quad \tilde{f} := \sup F(c);$
21: $\qquad\quad \mathsf{CutOffTest}\,(\mathscr{L}, \tilde{f});$
22: $\qquad F_Y := (F(c) + F'(Y_i)(Y_i - c)) \cap F(Y_i);$
23: $\qquad$ **if** $\inf F_Y \leqslant \tilde{f}$
24: $\qquad\quad$ **if** $d_{\mathrm{rel}}(F_Y) \leqslant \epsilon$ **or** $d_{\mathrm{rel}}(Y_i) \leqslant \epsilon$ **then**
25: $\qquad\qquad \mathscr{L}^* := \mathscr{L}^* \uplus (Y_i, \inf F_Y);$
26: $\qquad\quad$ **else**
27: $\qquad\qquad \mathscr{L} := \mathscr{L} \uplus (Y_i, \inf F_Y, F'(Y_i), c, F(c), \hat{f});$
28: $\quad$ **end for**
29: **end while**
30: $(Y, \inf F_Y) := \mathsf{Head}(\mathscr{L}^*); \quad F^* = [\inf F_Y, \tilde{f}];$
31: $\mathsf{CutOffTest}\,(\mathscr{L}^*, \tilde{f});$
32: **return** F^*, $\mathscr{L}^*$;

In Steps 4–7, we treat the boundary points separately and add them into the result list if they are candidates for minimizers. In Step 8, Algorithm 2 is called to outer prune the initial interval X using the values $f(\underline{x})$ and $f(\bar{x})$. A common lower bound $\hat{f}$ for the function values at the endpoints of the pruned interval is set in Step 9. We next bound the range of f over X according to the procedure described in Section 4.1 and add the pruned interval to the working list. Notice that each member of $\mathscr{L}$ is a structure that contains all necessary information associated with the current interval.

Steps 13–28 are applied to each candidate interval Y until the working list $\mathscr{L}$ is empty. The algorithm takes (and removes) the first element from $\mathscr{L}$ and prunes it according to the procedure described in Section 3.3. For each nonempty subinterval Y_i returned by the pruning step the monotonicity test is applied. When this test fails, a center c is selected for Y_i according to Algorithm 4. When the upper bound $\tilde{f}$ is changed, the cut-off test is applied to the working list $\mathscr{L}$ (Steps 19–21). After bounding the range of f over Y_i, the range test is applied in Step 23 and Y_i is added either to the result list $\mathscr{L}^*$ or to the working list $\mathscr{L}$ according to the termination criterion (Steps 24–27).

When no candidate interval is contained in the working list, the algorithm terminates by returning an enclosure for the global minimum F^* as well as all the elements of the result list $\mathscr{L}^*$. The next theorem establishes the correctness of Algorithm 5.

Theorem 13. *Let* $f : \mathscr{D} \to \mathbb{R}$ *be a* C^1 *function*, $X \subseteq \mathscr{D} \subseteq \mathbb{R}$, *and* $\epsilon > 0$. *Then Algorithm* 5 *has the following properties*:

1. $f^* \in F^*$.
2. $X^* \subseteq \bigcup_{(Y, \inf F_Y) \in \mathscr{L}^*} Y$.

Proof. Property 1 comes from the fact that $f^* = \min_{x \in X} f(x) \leqslant \tilde{f}$ while the elements of the result list $\mathscr{L}^*$ are sorted in a nondecreasing order with respect to the $\inf F_Y$. Property 2 is a consequence of the fact that no accelerating device used in the algorithm discards a global minimizer of f. $\square$

5. Numerical results

Algorithm 5 has been implemented and tested on an 846 MHz Pentium III with 256 MB RAM running Linux operating system. The implementation has been carried out in C++ using the C-XSC-2.0 library [13]. The inclusion functions have been implemented via the C-XSC routines while the derivatives are computed with forward automatic differentiation [9]. All computations, even the floating-point operations, have been performed using rounded interval arithmetic.

We compare our method with three already known methods that utilize the same type of information, i.e., objective and derivative inclusion function evaluations:

- The traditional method that uses monotonicity test, cut-off test and bisection as subdivision rule. For a fair numerical comparison, bounds have been obtained from the intersection of the mean value form with the natural interval extension, where the center of the form is the optimal center c^-.
- The kite method proposed by Vinko et al. [29], which uses the kite inclusion function that comes from the simultaneous usage of the mean value form and the linear boundary value form [19, p. 60]. The optimal center of the kite form has been approximated as suggested in [29] while the lower bound for the inclusion function is always taken as the best value between the infimum

of the natural interval extension and the kite inclusion form. Monotonicity test and cut-off test are also used in the kite method while as a subdivision rule we have adopted the split of the intervals at the approximated optimal center of the kite form. This approach avoids extra function evaluations at the midpoints of the intervals. As suggested in [29], pruning steps can be applied only at the end of the branching iteration. The exact way of applying them is not cleared and therefore, in our implementation pruning steps are not incorporated.
- The Casado et al. method, given in [7], where the authors suggest a new support function for the bounding process. Actually, the proposed support function is a modification of the linear boundary value form [19] without requiring the exact function values at the boundary points; instead, only lower bounds are needed (cf. [7, Theorem 3]). The method applies monotonicity test, cut-off test and bisection as a subdivision rule. Moreover, pruning is achieved through the gradient test described in [7].

All the above algorithms have been implemented and tested on 40 test problems given in [7]. [1] In all implementations a common stopping criterion has been adopted:

$$d_{\mathrm{rel}}(F_Y) \leqslant \epsilon \quad \text{or} \quad d_{\mathrm{rel}}(Y) \leqslant \epsilon,$$

with $\epsilon = 10^{-8}$. Comparison results between the traditional method (T), the kite method (K), the Casado et al. method (C) and the proposed method (P) are summarized in Table 1. For each test problem we report the number of function evaluations, the number of derivative evaluations, the number of subdivisions and the maximum list length. At the last two rows we sum the given indicators and report the relative compound indicators (as percents) for the three last methods (K, C, and P) compared with the traditional one (T), as explained before.

According to Table 1 our method exhibits the best performance among the kite and Casado et al. methods when compared with the traditional one. On total, we had 37% improvement in the number of function evaluations, 38% in the number of derivative evaluations and 10% improvement in the list length. The 87% improvement in the number of subdivisions reveals that our algorithm utilizes mostly inner pruning for the branching process rather than subdivision. Subdivision mainly takes place when the global minimum has been reached and the algorithm tries to isolate the global minimizer within an interval until the termination criterion is fulfilled.

[1] Executables and numerical results for all methods are available at `http://www.math.upatras.gr/~dgs/ugo/`.

Table 1
Comparison results on 40 test problems

No.	Function evaluation				Derivative evaluation				Subdivisions				List length			
	T	K	C	P	T	K	C	P	T	K	C	P	T	K	C	P
1	61	46	114	31	37	27	30	17	18	13	8	5	3	3	3	3
2	73	52	62	42	43	29	18	23	21	14	4	6	4	5	4	4
3	70	62	72	60	39	31	19	31	19	15	11	7	2	2	3	2
4	69	66	92	38	41	39	30	21	20	19	10	6	3	2	3	2
5	85	88	113	56	53	55	35	33	26	27	9	3	1	2	3	3
6	65	62	95	27	39	37	33	15	19	18	15	6	2	2	3	2
7	55	54	79	30	33	33	28	18	16	16	7	7	1	1	3	1
8	71	61	89	33	43	37	31	19	21	18	9	7	1	1	3	1
9	71	80	110	45	43	49	38	24	21	24	12	5	2	2	5	3
10	68	83	109	40	41	51	39	22	20	25	12	7	2	2	4	2
11	58	61	91	37	35	37	32	21	17	18	11	8	1	2	3	2
12	60	61	93	32	37	37	33	18	18	18	10	5	2	2	3	2
13	68	64	96	33	41	39	36	20	20	19	14	4	3	2	4	3
14	82	89	128	49	53	57	45	30	26	28	12	6	2	3	4	2
15	102	89	109	47	59	51	37	25	29	25	9	4	5	6	6	5
16	104	100	128	53	59	57	41	29	29	28	11	7	8	8	8	4
17	77	103	147	48	49	65	52	28	24	32	14	6	2	3	4	3
18	103	76	102	54	65	45	28	31	32	22	6	3	3	3	3	3
19	138	90	100	44	83	51	30	21	41	25	1	1	2	2	2	2
20	110	43	86	14	71	27	28	7	35	13	1	3	2	1	2	2
21	122	66	84	22	75	39	26	11	37	19	1	3	2	2	2	2
22	93	122	154	69	55	71	54	41	27	35	15	8	3	4	6	3
23	105	150	188	82	61	87	66	47	30	43	20	9	3	4	5	4
24	100	131	194	74	61	75	68	43	30	37	15	9	4	4	6	3
25	104	119	203	91	65	75	75	57	32	37	17	11	3	4	6	3
26	154	112	153	63	97	69	41	37	48	34	7	7	4	4	4	4
27	130	162	246	89	81	103	88	55	40	51	17	3	4	4	7	5
28	153	105	144	56	97	65	38	33	48	32	11	5	3	3	4	3

(*continued on next page*)

Table 1 (*continued*)

No.	Function evaluation				Derivative evaluation				Subdivisions				List length			
	T	K	C	P	T	K	C	P	T	K	C	P	T	K	C	P
29	131	164	226	95	81	93	80	55	40	46	16	6	5	5	6	4
30	211	148	211	80	135	93	55	49	67	46	14	5	4	4	5	4
31	93	111	131	78	53	65	42	42	26	32	13	9	4	5	8	4
32	412	283	425	173	263	177	111	105	131	88	10	7	8	8	9	8
33	298	279	387	181	173	161	129	103	86	80	22	8	15	16	17	15
34	171	253	322	126	95	131	114	71	47	65	21	10	5	6	9	5
35	233	263	346	177	125	135	114	97	62	67	12	10	7	6	12	7
36	428	491	708	319	213	245	240	175	106	122	19	8	11	9	16	10
37	226	295	398	148	125	157	143	82	62	78	27	11	9	8	14	6
38	436	476	636	310	279	243	226	184	139	121	36	8	24	10	18	10
39	1083	1188	1797	791	545	597	601	430	272	298	67	8	26	20	42	23
40	851	896	1383	650	425	447	456	339	212	223	54	9	25	19	42	25
$\sum$	7124	7244	10351	4487	4068	3982	3430	2509	2014	1971	600	260	220	199	311	199
		102%	145%	63%		98%	84%	62%		98%	30%	13%		90%	141%	90%

Total CPU time is an appropriate indicator for measuring the performance of the algorithms. The execution time (in seconds) of the above algorithms has been 0.72, 1.02, 0.92, and 0.45, respectively. Clearly, this shows the overall superiority of the proposed method, since it offers a substantial time saving.

6. Conclusion

In this work we present an interval branch-and-prune algorithm for computing verified enclosures for the global minimum and all global minimizers of univariate functions subject to bound constraints. The algorithm uses first-order information and utilizes a new pruning technique as well as a new branching strategy in order to accelerate the search process. Linear (inner and outer) pruning steps are responsible for the branching process and eliminate large parts of the search space where the global minimum does not exist. The algorithm exploits the attributes of the optimal center during the pruning and bounding process where either natural interval extension or optimal mean value form are used according to a selection criterion. We study valuable properties of the optimal center along with the mean value form and prove optimality with respect to the lower bound and to the width. Our algorithm has been implemented and tested on a standard set of 40 test functions and compared with the traditional, the kite, and the Casado et al. method. Numerical results show that the proposed method exhibits rich potentials and always outperforms the others.

Affine arithmetic [2] provides interval enclosures that are often narrower than those produced by interval arithmetic. Thus, it seems to be a possibility for further improving the efficiency of the proposed method by using affine arithmetic instead of interval arithmetic. See [18] and references therein for a detailed discussion on these issues.

Acknowledgement

The authors are greatly indebted to Dr. Elias C. Stavropoulos for his valuable suggestions that significantly improved the presentation of this work.

References

[1] G. Alefeld, J. Herzberger, Introduction to Interval Computations, Academic Press, London, 1983.

[2] M.V.A. Andrade, J.L.D. Comba, J. Stolfi. Affine arithmetic, in: Abstracts of the International Conference on Interval and Computer-Algebraic Methods in Science and Engineering (INTERVAL/94), St. Petersburg (Russia), March 1994, p. 36–40.

[3] E. Baumann, Globale Optimierung stetig differenzierbarer funktionen einer variablen (diplomarbeit), Freiburger Intervall-Berichte 86/6 (1986) 1–93.
[4] E. Baumann, Optimal centered forms, BIT 28 (1988) 80–87.
[5] O. Caprani, K. Madsen, Mean value forms in interval analysis, Computing 25 (1980) 147–154.
[6] L.G. Casado, J.A. Martínez, I. García, Experiments with a new selection criterion in a fast interval optimization algorithm, J. Global Optim. 19 (2001) 247–264.
[7] L.G. Casado, J.A. Martínez, I. García, Ya.D. Sergeyev, New interval analysis support functions using gradient information in a global minimization algorithm, J. Global Optim. 25 (4) (2003) 345–362.
[8] T. Csendes, New subinterval selection criteria for interval global optimization, J. Global Optim. 19 (2001) 307–327.
[9] R. Hammer, M. Hocks, U. Kulisch, D. Ratz, C++ Toolbox for Verified Computing I, Basic Numerical Problems: Theory, Algorithms, and Programs, Springer-Verlag, 1995.
[10] E. Hansen, Global Optimization Using Interval Analysis, Marcel Dekker, New York, 1992.
[11] E. Hansen, S. Sengupta, Global constrained optimization using interval analysis, in: K. Nickel (Ed.), Interval Mathematics, Springer-Verlag, Berlin, 1980, pp. 25–47.
[12] P. Hansen, B. Jaumard, J. Xiong, Cord-slope form of Taylor's expansion in univariate global optimization, J. Optim. Theor. Appl. 80 (3) (1994) 441–464.
[13] W. Hofschuster, W. Krämer. C-XSC 2.0–A C++ library for extended scientific computing. In: G. Goos, J. Hartmanis, J. van Leeuwen (Eds.), Numerical Software with Result Verification: International Dagstuhl Seminar, Dagstuhl Castle, Germany, January 19–24, 2003, Lecture Notes in Computer Science, vol. 2991, Springer, 2004, pp. 15–35.
[14] R.B. Kearfott, Rigorous Global Search: Continuous Problems, Kluwer Academic Publishers, Netherlands, 1996.
[15] R. Krawczyk, A. Neumaier, Interval slopes for rational functions and associated centered forms, SIAM J. Numer. Anal. 22 (1985) 604–616.
[16] R. Krawczyk, A. Neumaier, An improved interval Newton operator, J. Math. Anal. Appl. 118 (1986) 194–207.
[17] R.E. Moore, Interval Analysis, Prentice-Hall, Englewood Cliffs NJ, 1966.
[18] N.S. Nedialkov, V. Kreinovich, S.A. Starks, Interval arithmetic, affine arithmetic, Taylor series methods: why, what next? Numer. Algorithm 37 (1–4) (2004) 325–326.
[19] A. Neumaier, Interval Methods for Systems of Equations, Cambridge University Press, 1990.
[20] L.B. Rall, Mean value and Taylor forms in interval analysis, SIAM J. Math. Anal. 14 (2) (1983) 223–238.
[21] L.B. Rall, Representations of intervals and optimal error bounds, Math. Comput. 41 (163) (1983) 219–227.
[22] H. Ratschek, J. Rokne, Computer Methods for the Range of Functions, Ellis Horwood, Chichester, 1984.
[23] H. Ratschek, J. Rokne, New Computer Methods for Global Optimization, Ellis Horwood, Chichester, 1988.
[24] H. Ratschek, J. Rokne, Formulas for the width of interval products, Reliable Comput. 1 (1) (1995) 9–14.
[25] D. Ratz, Automatic Slope Computation and its Application in Nonsmooth Global Optimization, Shaker Verlag, Aachen, 1998.
[26] D. Ratz, A nonsmooth global optimization technique using slopes: the one-dimensional case, J. Global Optim. 14 (1999) 365–393.
[27] D.G. Sotiropoulos, T.N. Grapsa, A branch-and-prune method for global optimization: the univariate case, in: W. Kräemer, J.W. von Gudenberg (Eds.), Scientific Computing, Validated Numerics, Interval Methods, Kluwer Academic/Plenum Publishers, New York, 2001, pp. 215–226.

[28] D.G. Sotiropoulos, T.N. Grapsa. An interval branch and bound algorithm for global optimization using pruning steps, in: E.A. Lipitakis (Ed.), Proceedings of the Fifth Hellenic-European Conference on Computer Mathematics and its Applications (HERCMA 2001), vol. 2, LEA Press, 2002, pp. 455–459.
[29] T. Vinko, J.-L. Lagouanelle, T. Csendes. A new inclusion function for optimization: Kite—the one dimensional case. J. Global Optim., in press.