On the Complexity of Exclusion Algorithms for Optimization

Eugene Allgower,[1] Melissa Erdmann, and Kurt Georg[1,2]

Department of Mathematics, Colorado State University, Fort Collins, Colorado 80523

Exclusion algorithms are a well-known tool in the area of interval analysis for finding all solutions of a system of nonlinear equations or for finding the global minimum of a function over a compact domain. The present paper discusses a new class of tests for such algorithms in the context of global optimization and presents complexity results concerning the resulting algorithms.   © 2002 Elsevier Science (USA)

1. INTRODUCTION

Exclusion algorithms are a well-known tool in the area of interval analysis, see, e.g., [6, 7], for finding all solutions of a system of nonlinear equations or for finding the global minimum of a function over a compact domain. They also have been introduced in [11, 12] from a slightly different viewpoint. In particular, such algorithms seem to be very useful for finding all solutions of low-dimensional, but highly nonlinear systems which have many solutions or for finding the global minimum of a function of a few variables in the presence of many local minima. Such systems occur, e.g., in mechanical engineering.

In [4], the third author introduced some new exclusion tests and analyzed the efficiency and computational complexity of exclusion algorithms based on this approach, for the case of finding all zero points of a system of nonlinear equations. The present paper investigates the complexity of exclusion algorithms for finding the global minimum of a function, where the exclusion tests are suitable modifications of the ones obtained in [4].

573

We briefly describe our view of the exclusion method.

In $\mathbb{R}^n$ and $\mathbb{R}^{m \times n}$ we use the component-wise $\leqslant$ as a partial ordering, and $|\cdot|$ is the component-wise absolute value. We only use the max norm $\|\cdot\|$. For example, for two matrices $A, B \in \mathbb{R}^{n \times n}$ the symbol $A \leqslant B$ means that $A(i, j) \leqslant B(i, j)$ for $i, j = 1 : n$.

An interval σ in $\mathbb{R}^n$ is a rectangular box, i.e., there are two vectors m_σ, $r_\sigma \in \mathbb{R}^n$ with $r_\sigma(i) > 0$, $i = 1 : n$, such that

$$\sigma = [m_\sigma - r_\sigma, m_\sigma + r_\sigma] = \{x \in \mathbb{R}^n : m_\sigma - r_\sigma \leqslant x \leqslant m_\sigma + r_\sigma\}.$$

We call $m_\sigma - r_\sigma$ the lower corner, $m_\sigma + r_\sigma$ the upper corner, m_σ the midpoint, and r_σ the radius of σ. (This corresponds to the midpoint–radius representation in interval analysis.)

Let $\sigma \subset \mathbb{R}^n$ be an interval and $F \colon \sigma \to \mathbb{R}^n$ a function defined on σ. We call a test

$$T_F(\sigma) \in \{0, 1\} \qquad \text{where} \quad 0 \equiv \text{no} \quad \text{and} \quad 1 \equiv \text{yes}$$

an *exclusion test* for F on σ iff $T_F(\sigma) = 0$ implies that F has no zero point in σ. Hence, $T_F(\sigma) = 1$ is a *necessary* condition for F to have a zero point in σ.

This notion is strongly reminiscent of the inclusion test introduced in an abstract setting in [5]. It seems that the notion and use of exclusion tests goes at least back to Moore; see [9, $E(X)$ on p. 77].

If an exclusion test is given, then we can recursively bisect intervals and discard the ones which yield a negative test. This leads to the following recursive *exclusion algorithm* which we start from some initial interval Λ on which F is defined. We assume that an exclusion test $T_F(\sigma)$ is available for all subintervals $\sigma \subset \Lambda$.

ALGORITHM 1 (EXCLUSION ALGORITHM).
$\Gamma \leftarrow \{\Lambda\}$ (initial interval)
for $\ell = 1 :$ `maximal_level`
   for $a = 1 : n$
     let $\tilde{\Gamma}$ be obtained by bisecting each $\sigma \in \Gamma$ along the axis a
     for $\sigma \in \tilde{\Gamma}$
       if $T_F(\sigma) = 0$
         drop σ from $\tilde{\Gamma}$ (σ is excluded)
     $\Gamma \leftarrow \tilde{\Gamma}$
   $\Gamma_\ell \leftarrow \Gamma$ (for later reference)

Remark 2. The exclusion algorithm is similar to early algorithms in interval analysis. It turns out that bisection is an efficient partitioning

strategy. In order to simplify and unify our efficiency investigations, we have considered only the strategy of cyclic bisections of the intervals along subsequent axes. Various authors have investigated bisection schemes. For a fairly early discussion see [9, pp. 77–81]. For a further careful comparison of bisection schemes, see [2]. This will be further investigated in [3].

For clarity of exposition and notation, the list of intervals is processed breadth-first rather than depth-first. However, we mention that the other choice (which uses less memory) was actually implemented in [4]. It is easy to see that the complexity analysis presented in [4] is not influenced by this difference in choice. We refer also to the analysis appearing in [10, pp. 77–80, 85–102].

However, let us already point out here that for the present paper the choice between breadth-first and depth-first is crucial; see Remark 5.

Whenever one cycle of bisections is accomplished, we say that we have reached a new bisection level, and we think of an exclusion algorithm as performing a fixed number of bisection levels. The intervals which have not been discarded after ℓ bisection levels will be considered as the intervals which the algorithm generates on the ℓth bisection level; see Fig. 1 for an illustration. The list of these intervals is denoted by Γ_ℓ in the algorithm. Obviously, if $\Gamma_\ell = \varnothing$ for some level ℓ, then the algorithm has shown that there are no zero points of F in the initial interval Λ.

It is clear that the efficiency of exclusion algorithms hinges mainly on the construction of a good exclusion test which is computationally inexpensive but relatively tight. Otherwise, too many intervals remain undiscarded on each bisection level, and this leads to significant numerical inefficiency.

In the area of interval analysis, the idea of exclusion is exploited in *interval branch and bound algorithms* which are used to find all the zero points of a nonlinear system of equations or also to minimize functions; see, e.g., Kearfott [7] and the references cited therein, and the software

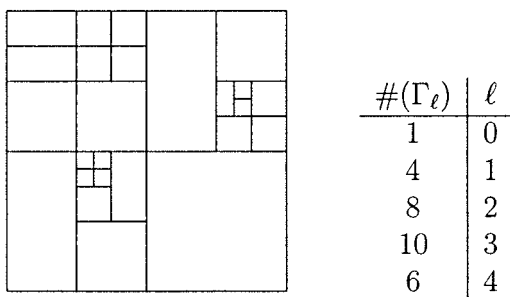| $\#(\Gamma_\ell)$	ℓ
1	0
4	1
8	2
10	3
6	4

FIG. 1. Illustration of bisection levels.

package GlobSol accompanying the book [6]. From an interval analysis viewpoint, a simple exclusion test could be designed in the following way,

$$T_f(\sigma) = 1 :\Leftrightarrow 0 \in [f](\sigma),$$

where $[f](\sigma)$ is the interval obtained from σ by applying f in an interval analysis sense. More sophisticated tests employ an interval-Newton step.

The following complexity result was shown in [11, 12] for a certain class of exclusion tests:

THEOREM 3. *Let* $\Lambda \subset \mathbb{R}^n$ *be an interval, and let* $F: \Lambda \to \mathbb{R}^n$ *be sufficiently smooth and zero a regular value of* F. *Then there is a constant* $C > 0$ *such that the exclusion algorithm, started in* Λ, *generates no more than* C *intervals on each bisection level; i.e.,* $\#(\Gamma_\ell) \leqslant C$ *independent of* ℓ.

A related analysis concerning clustering of undiscarded intervals on various levels as a function of the sharpness of the lower bound on the range was given in [8].

Hence, if the complexity of one exclusion test is known, then this leads immediately to a complexity statement on the efficiency of an exclusion algorithm. However, the constant C could be very big, and numerical experiments, see [4], show that the efficiency of exclusion algorithms varies considerably with the choice of the test. This is of particular importance for more demanding nonlinear systems, such as those which typically occur in engineering.

Furthermore, some applications lead to nonlinear systems where some of the zero points are singular. In this case, Theorem 3 cannot be applied, and in fact is wrong, as some examples in [4] show. Therefore, for a certain improved class of exclusion tests, the assumptions in Theorem 3 were considerably relaxed in [4].

The aim of this paper is to begin a similar investigation for optimization codes. It will turn out that the techniques used in [4] cannot be simply translated into the new situation, but a significantly different analysis has to be performed.

2. EXCLUSION ALGORITHMS FOR UNCONSTRAINED MINIMIZATION

Let $\sigma \subset \mathbb{R}^n$ be an interval and $f: \sigma \to \mathbb{R}$ a function defined on σ. We call a number $E_f(\sigma) \in \mathbb{R}$ a *lower bound* of f on σ iff

$$\inf_{x \in \sigma} f(x) \geqslant E_f(\sigma).$$

Note that $E_f(\sigma)$ is also a lower bound (possibly not a good one) of f on τ if $\tau \subset \sigma$ is a subinterval.

If such bounds are available, then we can recursively bisect intervals and discard the ones whose lower bounds are larger than a currently found low value M of f. This leads to the following recursive *exclusion algorithm* which we start from some initial interval $\Lambda \subset \mathbb{R}^n$ on which f is defined. We assume that a lower bound $E_f(\sigma)$ is available for all subintervals $\sigma \subset \Lambda$.

ALGORITHM 4 (EXCLUSION ALGORITHM FOR MINIMIZATION).
$\quad \Gamma \leftarrow \{\Lambda\}$ (initial interval)
$\quad M \leftarrow f(m_\Lambda)$ (initial low value of f)
$\quad$ for $\ell = 1 :$ max_level
$\quad\quad$ for $a = 1 : n$
$\quad\quad\quad$ let $\tilde{\Gamma}$ be obtained by bisecting each $\sigma \in \Gamma$ along the axis a
$\quad\quad\quad$ for $\sigma \in \tilde{\Gamma}$
$\quad\quad\quad\quad$ if $E_f(\sigma) > M$
$\quad\quad\quad\quad\quad$ drop σ from $\tilde{\Gamma}$
$\quad\quad\quad\quad$ else
$\quad\quad\quad\quad\quad$ $M \leftarrow \min(M, f(m_\sigma))$ (update)
$\quad\quad\quad$ $\Gamma \leftarrow \tilde{\Gamma}$
$\quad\quad$ $\Gamma_\ell \leftarrow \Gamma, M_\ell \leftarrow M$ (for later reference)

Remark 5. The preceding algorithm processes the list of intervals breadth-first rather than depth-first. Contrary to Algorithm 1, this distinction is now very important. The reason is that for our upcoming complexity results the global variable M needs to be updated uniformly over the initial interval Λ.

Let us draw some immediate conclusions from the design of the above algorithm:

LEMMA 6. $\Gamma_\ell \neq \varnothing$ for all ℓ.

LEMMA 7. *The sequence* M_ℓ *is monotonically decreasing, and* $\inf_{x \in \Lambda} f(x) \leqslant M_\ell \leqslant \min_{\sigma \in \Gamma_\ell} f(m_\sigma)$.

LEMMA 8. *Let* $\sigma \in \Gamma_\ell$. *Then*

1. $E_f(\sigma) \leqslant M$ *where* $M_{\ell-1} \geqslant M \geqslant M_\ell$,
2. $M_\ell \leqslant f(m_\sigma)$,
3. $\|r_\sigma\| = 2^{-\ell} r_\Lambda$.

LEMMA 9. *If* $\xi \in \Lambda$ *such that* $f(\xi) = \inf_{x \in \Lambda} f(x)$, *then* $\xi \in \bigcup_{\sigma \in \Gamma_\ell} \sigma$ *for each* ℓ.

LEMMA 10. *If f is continuous on Λ, then*

$$\lim_{\ell \to \infty} M_t = \min_{x \in \Lambda} f(x).$$

Proof. Since f is continuous, there is a $\xi \in \Lambda$ such that $f(\xi) = \inf_{x \in \Lambda} f(x)$. By Lemma 9 we know that there is a sequence σ_ℓ with $\xi \in \sigma_\ell \in \Gamma_\ell$. Note that $\lim_{\ell \to \infty} m_{\sigma_\ell} = \xi$. By Lemma 7 we know that $f(\xi) \leqslant M_\ell \leqslant f(m_{\sigma_\ell})$ and $\lim_{\ell \to \infty} f(m_{\sigma_\ell}) = f(\xi)$ concludes the proof. ∎

Again, it is clear that the efficiency of the above exclusion algorithm hinges mainly on the construction of good lower bounds which are computationally inexpensive but relatively tight. Otherwise, too many intervals remain undiscarded on each bisection level, and this leads to significant numerical inefficiency. It is natural to involve higher derivatives in such bounds; see also [1].

In order to obtain refined lower bounds $E_f(\sigma)$, in this paper we will use the construction of dominant functions introduced in [4]. Let us briefly review this concept in the next section.

3. CONSTRUCTION OF DOMINANT FUNCTIONS

We denote by $\mathbb{Z}_+$ the set of nonnegative integers. For a multi-index

$$\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{Z}_+^n$$

we consider the following definitions:

1. The length of α is defined by $|\alpha| := \sum_i \alpha_i$.
2. The factorial of α is defined by $\alpha! := \prod_i \alpha_i!$.
3. If $x \in \mathbb{R}^n$, then we define $x^\alpha := \prod_i x_i^{\alpha_i}$.
4. We define the partial derivatives $\partial^\alpha = (\alpha!)^{-1} \prod_i \partial_i^{\alpha_i}$.

Furthermore, we introduce the probability measures

$$\omega_k(dt) = k(1-t)^{k-1}\, dt$$

on the interval $[0, 1]$.

Using these definitions, Taylor's formula with $k > 0$ and integral remainder is easy to write:

$$f(m+h) = f(m) + \sum_{0 < |\alpha| < k} \partial^\alpha f(m)\, h^\alpha$$

$$+ \sum_{|\beta| = k} \int_0^1 \partial^\beta f(m+th)\, \omega_k(dt)\, h^\beta. \tag{11}$$

DEFINITION 12.   Let $\sigma \subset \mathbb{R}^n$ be an interval. By $\mathscr{A}_k(\sigma)$ we denote the space of functions $f: \sigma \to \mathbb{R}$ such that $\partial^\alpha f$ is absolutely continuous for $|\alpha| < k$. Note that for $f \in \mathscr{A}_k$ the Taylor formula (11) holds. In $\mathscr{A}_k(\sigma)$ we introduce the cone

$$\mathscr{K}_k(\sigma) = \{g \in \mathscr{A}_k(\sigma): 0 \leqslant \partial^\alpha g(x) \leqslant \partial^\alpha g(y) \text{ for } 0 \leqslant x \leqslant y, |\alpha| \leqslant k\}.$$

We also set

$$\mathscr{A}_\infty(\sigma) := \bigcap_{k=1}^\infty \mathscr{A}_k(\sigma) \qquad \text{and} \qquad \mathscr{K}_\kappa(\sigma) := \bigcap_{k=1}^\infty \mathscr{K}_k(\sigma).$$

We now introduce the notion of a dominant function which will be the basis for the lower bounds considered in this paper:

DEFINITION 13.   Let $f \in \mathscr{A}_k(\sigma)$ and $g \in \mathscr{K}_k(\sigma)$. Then $f(x) \prec_k g(x)$ for $x \in \sigma$ (g dominates f with order k on σ) iff the estimates

$$|\partial^\alpha f(x)| \leqslant \partial^\alpha g(|x|)$$

hold for all $x \in \sigma$ and $|\alpha| \leqslant k$. If $f \in \mathscr{A}_\infty(\sigma)$ and $g \in \mathscr{K}_\infty(\sigma)$, then $f(x) \prec_\infty g(x)$ for $x \in \sigma$ means that $f \prec_k g$ for $x \in \sigma$ and all $k \geqslant 0$.

Note that $f(x) \prec_k g(x)$ for $x \in \sigma$ obviously implies that $f(x) \prec_q g(x)$ for $x \in \tau$, provided that $q \leqslant k$ and $\tau \subset \sigma$. We will frequently use the notation $f \prec_k g$ or $f(x) \prec_k g(x)$ if there is no ambiguity about the underlying interval.

The following examples point out the differences between the various estimates:

EXAMPLE 14.

   1.   If $g \in \mathscr{K}_k$ then $g \prec_k g$. This includes examples such as $\exp(m+x) \prec_\infty \exp(m+x)$, and $\tan x \prec_\infty \tan x$ for $|x| < \frac{\pi}{2}$.

   2. $\sin x \prec_\infty \sinh x$, but $\sin x \prec_3 x + \frac{1}{6} x^3$.

   3. $\cos x \prec_\infty \cosh x$, but $\cos x \prec_1 1 + x$, $\cos x \prec_2 1 + \frac{1}{2} x^2$, and $\cos x \prec_3 1 + \frac{1}{2} x^2 + \frac{1}{6} x^3$.

   4. $\log(1+x) \prec_\infty \log(1-x)$ but $\log(1+x) \prec_3 x + \frac{1}{2} x^2 + \frac{1}{3} x^3$ for $|x| < 1$.

   5. $\sin(m+x) \prec_\infty \sinh(|m|+x)$ but $\sin(m+x) \prec_2 |\sin(m)| + |\cos(m)| x + \frac{1}{2} x^2$.

We list some rules, see [4], that can be used as a tool to generate dominant functions in much the same way as rules about differentiation are used as a tool to generate derivatives.

THEOREM 15.

1. $f \prec_k g$ implies $f(m+x) \prec_k g(|m|+x)$.
2. $f \prec_1 g$ implies $|f| \prec_1 g$.
3. Let $f \prec_k g$ and $\lambda \in \mathbb{R}$. Then $\lambda f \prec_k |\lambda| g$.
4. Let $f_i \prec_k g_i$, $i = 1 : q$. Then $\sum_i f_i \prec_k \sum_i g_i$.
5. Let $f_i \prec_k g_i$, $i = 1, ..., q$. Then $\prod_i f_i \prec_k \prod_i g_i$.
6. Let $f \prec_k g$ and $f_i \prec_k g_i$, $i = 1, ..., n$. Set $F = f(f_1, ..., f_n)$ and $G = g(g_1, ..., g_n)$. Then $F \prec_k G$.

Here are some examples of how the preceding rules could be applied:

EXAMPLE 16.

1. $e^{|\sin(m+x)|} \prec_1 e^{|\sin m| + x}$.
2. $\frac{1}{1+t} \prec_\infty \frac{1}{1-t}$ for $|t| < 1$ and $\sin(x) \prec_3 x + \frac{1}{6} x^3$ implies

$$\frac{1}{1 + \frac{1}{2}\sin(x)} \prec_3 \frac{1}{1 - \frac{1}{2}(x + \frac{1}{6} x^3)} \quad \text{for} \quad |x + \frac{1}{6} x^3| < 2.$$

3. $\sin(x_1^2)\cos(x_2 - x_3) \prec_2 (x_1^2 + \frac{1}{2}(x_1^2)^2)(1 + \frac{1}{2}(x_2 + x_3)^2)$.

4. LOCAL EXPANSIONS TO OBTAIN LOWER BOUNDS

The following theorem summarizes the possible choices of lower bound estimates which we will consider in this paper for Algorithm 4:

THEOREM 17. Let $\sigma \subset \mathbb{R}^n$ be an interval, and let $q > 0$ be an integer. Let $f(m_\sigma + x) \prec_q g(x)$ for $|x| \leqslant r_\sigma$. Then

$$E_f(\sigma) := f(m_\sigma) - \underbrace{(g(r_\sigma) - g(0))}_{\geqslant 0} + \sum_{0 < |\alpha| < q} \underbrace{(\partial^\alpha g(0) - |\partial^\alpha f(m_\sigma)|)}_{\geqslant 0} r_\sigma^\alpha \qquad (18)$$

is a lower bound for f on σ.

Proof. Let $m_\sigma + h \in \sigma$ such that $f(m_\sigma + h) = \min_{x \in \sigma} f(x)$. We have to show that $f(m_\sigma + h) \geqslant E_f(\sigma)$. Using the Taylor formula (11) we obtain

$$g(r_\sigma) = g(0) + \sum_{0 < |\alpha| < q} \partial^\alpha g(0) \, r_\sigma^\alpha + \int_0^1 \sum_{|\beta| = q} \partial^\beta g(t r_\sigma) \, \omega_q(dt) \, r_\sigma^\beta$$

and consequently

$$f(m_\sigma + h) = f(m_\sigma) + \sum_{0 < |\alpha| < q} \partial^\alpha f(m_\sigma)\, h^\alpha + \int_0^1 \sum_{|\beta| = q} \partial^\beta f(m_\sigma + th)\, \omega_q(dt)\, h^\beta$$

$$\geqslant f(m_\sigma) - \sum_{0 < |\alpha| < q} |\partial^\alpha f(m_\sigma)\, h^\alpha| - \int_0^1 \sum_{|\beta| = q} |\partial^\beta f(m_\sigma + th)\, \omega_q(dt)\, h^\beta|$$

$$\geqslant f(m_\sigma) - \sum_{0 < |\alpha| < q} |\partial^\alpha f(m_\sigma)|\, r_\sigma^\alpha - \int_0^1 \sum_{|\beta| = q} \partial^\beta g(t r_\sigma)\, \omega_q(dt)\, r_\sigma^\beta$$

$$= f(m_\sigma) - (g(r_\sigma) - g(0)) + \sum_{0 < |\alpha| < q} (\partial^\alpha g(0) - |\partial^\alpha f(m_\sigma)|)\, r_\sigma^\alpha.$$

COROLLARY 19.   *Let $\sigma \subset \mathbb{R}^n$ be an interval, and let $q > 0$ be an integer. Let $f(x) \prec_q g(x)$ for $x \in \sigma$. Then*

$$E_f(\sigma) := f(m_\sigma) - (\underbrace{g(|m_\sigma| + r_\sigma) - g(|m_\sigma|)}_{\geqslant 0})$$

$$+ \sum_{0 < |\alpha| < q} \underbrace{(\partial^\alpha g(|m_\sigma|) - |\partial^\alpha f(m_\sigma)|)}_{\geqslant 0}\, r_\sigma^\alpha \qquad (20)$$

is a lower bound for f on σ.

Proof.   Note that $f(m_\sigma + x) \prec_q g(|m_\sigma| + x)$ for $|x| \leqslant r_\sigma$ and apply the theorem.   ∎

The terms inside the summation sign in (18) and (20) are nonnegative, and hence the estimate tightens with increasing q; i.e., the lower bound estimate increases. To increase the efficiency of implementations, one would successively apply the estimate for $q = 1 : q_0$ in Algorithm 4 and discard the interval as soon as possible.

5. COMPLEXITY RESULTS

In this section we investigate the complexity of Algorithm 4 in the sense of Theorem 3.

Throughout this section, let $\Lambda \subset \mathbb{R}^n$ be an initial interval, $q > 0$ an integer, $f : \Lambda \to \mathbb{R}$, and $f(x) \prec_q g(x)$ for $x \in \Lambda$. We start Algorithm 4 in Λ using the lower bound estimate (20) for all subintervals $\sigma \subset \Lambda$. Recall that Algorithm 4 generates for each level $i > 0$ a list of intervals Γ_i. For the purpose of an asymptotic analysis, we assume that `maximal_level` $= \infty$; i.e., we consider the algorithm to run without termination.

We introduce the set of minimal points

$$\mathcal{M}_f(\Lambda) := \{\xi \in \Lambda \colon f(x) = \min_{x \in \Lambda} f(x)\}.$$

Note that $\mathcal{M}_f(\Lambda) \neq \varnothing$. We will need the following technical definition:

DEFINITION 21. We say that $\xi \in \mathcal{M}_f(\Lambda)$ has uniform order p if

1. $\partial^\alpha f(\xi) = 0$ for $0 < |\alpha| \leqslant p$,

2. there exists an $\varepsilon > 0$ such that $\varepsilon \|m - \xi\|^{p+1} \leqslant f(m) - f(\xi)$ for $\|m - \xi\| \leqslant \varepsilon$.

We recall that a point $\xi \in \mathcal{M}_f(\Lambda)$ which is also in the interior of Λ satisfies the (necessary) minimality conditions

1. $\partial^\alpha F(\xi) = 0$ for $|\alpha| = 1$.

2. The quadratic form $\sum_{|\beta| = 2} \partial^\beta f(\xi)\, x^\beta$ is positive semidefinite.

Here we have to assume that f is sufficiently smooth; say, $f \in \mathcal{A}_3(\Lambda)$.

If furthermore ξ is a regular point of the gradient of f (the generic case), then the quadratic form is positive definite, and Taylor's formula implies immediately that ξ has uniform order 1. Hence the previous definition (for $p > 1$) allows some singularity. The following lemma is the basis for our complexity analysis.

LEMMA 22. *Let each point in $\mathcal{M}_f(\Lambda)$ have some uniform order p with $p < q$. Then there exists an $A > 0$ such that the following holds: if $\sigma \in \Gamma_\ell$ with $\ell > A$, then there exists a $\xi \in \mathcal{M}_f(\Lambda)$ such that $\|m_\sigma - \xi\| \leqslant A \|r_\sigma\|$.*

Proof. Assume not. Then there is a sequence $\sigma_i \in \Gamma_i$ of intervals such that $\|r_{\sigma_i}\| \leqslant 1/i$ and

$$\|m_{\sigma_i} - \eta\| > i \|r_{\sigma_i}\| \qquad \text{for all} \quad \eta \in \mathcal{M}_f(\Lambda). \tag{23}$$

Since Λ is compact, we find a convergent subsequence of the m_{σ_i}; i.e., there is an unbounded set I of natural numbers such that

$$\lim_{i \in I} m_{\sigma_i} = \xi$$

for some $\xi \in \Lambda$. From Lemma 8 and from (20) it follows that

$$f(m_{\sigma_i}) - (g(|m_{\sigma_i}| + r_{\sigma_i}) - g(|m_{\sigma_i}|)) \\ + \sum_{0 < |\alpha| < q} (\partial^\alpha g(|m_{\sigma_i}|) - |\partial^\alpha f(m_{\sigma_i})|)\, r_{\sigma_i}^\alpha \leqslant M_{i-1}.$$

Letting $i \to \infty$ while $i \in I$, from Lemma 10 we obtain that $\xi \in \mathscr{M}_f(\Lambda)$. By assumption we know that ξ has a certain uniform order, say p, with $p < q$. Since the estimate (20) is increasing for increasing q, we have

$$f(m_{\sigma_i}) - (g(|m_{\sigma_i}| + r_{\sigma_i}) - g(|m_{\sigma_i}|))$$
$$+ \sum_{0 < |\alpha| \leqslant p} (\partial^\alpha g(|m_{\sigma_i}|) - |\partial^\alpha f(m_{\sigma_i})|)\, r_{\sigma_i}^\alpha \leqslant M_{i-1}.$$

Since

$$g(|m_{\sigma_i}| + r_{\sigma_i}) = g(|m_{\sigma_i}|) + \sum_{0 < |\alpha| \leqslant p} \partial^\alpha g(|m_{\sigma_i}|)\, r_{\sigma_i}^\alpha + \mathbf{O}(\|r_{\sigma_i}\|^{p+1})$$

we have

$$f(m_{\sigma_i}) + \sum_{0 < |\alpha| \leqslant p} |\partial^\alpha f(m_{\sigma_i})|\, r_{\sigma_i}^\alpha \leqslant M_{i-1} + \mathbf{O}(\|r_{\sigma_i}\|^{p+1}). \qquad (24)$$

Expanding $\partial^\alpha f(m_{\sigma_i})$ about ξ and using the fact that all derivatives of order $\leqslant p$ vanish, we obtain

$$\partial^\alpha f(m_{\sigma_i}) = \sum_{\gamma:\, |\gamma| + |\alpha| = p+1} \int_0^1 \partial^\gamma \partial^\alpha f(\xi + t(m_{\sigma_i} - \xi))\, \omega_{p+1-|\alpha|}(dt)(m_{\sigma_i} - \xi)^\gamma.$$

This leads to

$$|\partial^\alpha f(m_{\sigma_i})|\, r_{\sigma_i}^\alpha = \mathbf{O}(\|r_{\sigma_i}\|^{|\alpha|}\, \|m_{\sigma_i} - \xi\|^{p+1-|\alpha|}).$$

Using this and (23) in (24) gives

$$f(m_{\sigma_i}) \leqslant M_{i-1} + \mathbf{O}(\|r_{\sigma_i}\|\, \|m_{\sigma_i} - \xi\|^p).$$

On the other hand, for $i \in I$ there is a $\tau_{i-1} \in \Gamma_{i-1}$ such that $\xi \in \tau_{i-1}$. Hence, using the Taylor formula and Lemma 8 we see that

$$M_{i-1} \leqslant f(m_{\tau_i}) = f(\xi) + \mathbf{O}(\|r_{\tau_{i-1}}\|^{p+1}) = f(\xi) + \mathbf{O}(\|r_{\sigma_i}\|^{p+1}).$$

The last two inequalities now lead to

$$f(m_{\sigma_i}) - f(\xi) \leqslant \mathbf{O}(\|r_{\sigma_i}\|\, \|m_{\sigma_i} - \xi\|^p).$$

Since ξ has uniform order p, there is an $\varepsilon > 0$ such that

$$\varepsilon \, \|m_{\sigma_i} - \xi\|^{p+1} \leqslant f(m_{\sigma_i}) - f(\xi)$$

for all but finitely many $i \in I$. Combining the last two inequalities leads to

$$\varepsilon \, \|m_{\sigma_i} - \xi\|^{p+1} \leqslant \mathbf{O}(\|r_{\sigma_i}\| \, \|m_{\sigma_i} - \xi\|^p)$$

for all but finitely many $i \in I$, clearly in contradiction to (23). ∎

The proof of the following theorem is now simple, but somewhat technical in its precise details; see also [4].

THEOREM 25.   *Let each point in $\mathscr{M}_f(\Lambda)$ have some uniform order p with $p < q$. Then $\#\Gamma_\ell$ is bounded as $\ell \to \infty$.*

Proof. Given the radius r_Λ of the initial interval Λ, let $\eta :=$ $\min_v r_\Lambda(v) > 0$ be its minimal entry. Let e denote the vector with all entries equal to 1. Let A be the constant of the previous lemma. We only need to consider bisection levels $\ell > A$.

Let $\sigma \in \Gamma_\ell$, and let $\xi \in \mathscr{M}_f(\Lambda)$ be such that $\|m_\sigma - \xi\| \leqslant A \, \|r_\sigma\|$. Note that we can write this inequality as

$$\xi - A \, \|r_\sigma\| \, e \leqslant m_\sigma \leqslant \xi + A \, \|r_\sigma\| \, e.$$

Note that $r_\sigma = 2^{-\ell} r_\Lambda$. From $e \leqslant r_\Lambda / \eta$ it follows that

$$\xi - \frac{A \, \|r_\Lambda\|}{\eta} r_\sigma = \xi - \frac{A \, \|r_\sigma\|}{\eta} r_\Lambda \leqslant \xi - A \, \|r_\sigma\| \, e$$

$$\leqslant m_\sigma \leqslant \xi + A \, \|r_\sigma\| \, e \leqslant \xi + \frac{A \, \|r_\sigma\|}{\eta} r_\Lambda = \xi + \frac{A \, \|r_\Lambda\|}{\eta} r_\sigma.$$

Hence, if L is an integer such that

$$L \geqslant \frac{A \, \|r_\Lambda\|}{\eta} + 1,$$

then σ is contained in the interval $\tau_\xi = [\xi - L r_\sigma, \, \xi + L r_\sigma]$. There are at most L^n intervals in Γ_ℓ that can be contained in τ_ξ. Since $\mathscr{M}_f(\Lambda)$ is compact, and the assumptions imply that the points in $\mathscr{M}_f(\Lambda)$ are isolated, we have $\#\mathscr{M}_f(\Lambda) \leqslant C$ for some $C > 0$, and hence $\#\Gamma_\ell \leqslant L^n C$. ∎

Remark 26.   Not all isolated zero points of the gradient of f, even if f is analytic, satisfy Definition 21 for some p; in fact, orders of such zero

points are defined in a different way. Modifications of the above proof for more general cases will be investigated elsewhere.

Remark 27.   The preceding theorem indicates that Algorithm 4 should be used with an estimate (20) where $q \geqslant 2$, i.e., should involve at least first-order derivatives of f, in order to avoid a blow-up in the number of intervals per bisection level.

6. SPECIAL CASE: POLYNOMIAL SYSTEMS

For polynomial systems it is natural to use the following simple dominance. Given a polynomial of degree r

$$p(x) = \sum_{|\alpha| \leqslant r} c_\alpha x^\alpha,$$

we define

$$\hat{p}(x) = \sum_{|\alpha| \leqslant r} |c_\alpha| \, x^\alpha,$$

and therefore have $p \prec_\infty \hat{p}$. The estimate (20) now reads

$$E_f(\sigma) := p(m_\sigma) - (\underbrace{\hat{p}(|m_\sigma| + r_\sigma) - \hat{p}(|m_\sigma|)}_{\geqslant 0})$$

$$+ \sum_{0 < |\alpha| < q} (\underbrace{\partial^\alpha \hat{p}(|m_\sigma|) - |\partial^\alpha p(m_\sigma)|}_{\geqslant 0}) \, r_\sigma^\alpha. \tag{28}$$

A numerically important observation is that under certain conditions the terms in the above sum are zero. More precisely:

DEFINITION 29.   We call a polynomial p monotone iff all nonzero coefficients of p have the same sign.

The following two lemmas are rather obvious:

LEMMA 30.   *A polynomial p is monotone iff $\hat{p}(|m|) = |p(m)|$ for all $m \geqslant 0$.*

LEMMA 31.   *If p is monotone, then $\partial^\beta p$ is monotone for all β.*

The case when our initial interval Λ is in the positive cone is an important one. Often for systems with physical significance, variables only take on positive values. Then the preceding observations enable us to identify the multi-indices α, for which the summation in (28) needs to be carried

out. The following recursion generates these multi-indices in an efficient way:

```
function GenerateMultiIndices(α)
set n = |α|
if ∂^α(p) is monotone
   return
print(α)
set β = α
set β_1 = β_1 + 1
GenerateMultiIndices(β)
for k = 1 : n − 1
   if α_k ≠ 0
      return
   set β = α
   set β_{k+1} = β_{k+1} + 1
   GenerateMultiIndices(β)
```

The recursion is started with $\alpha = (0, \ldots, 0)$.

On the other hand, for $q = \infty$ in (28), we obtain a simplification:

$$E_f(\sigma) := p(m_\sigma) - \sum_{0 < |\alpha| < q} \partial^\alpha |\partial^\alpha p(m_\sigma)| \, r_\sigma^\alpha. \tag{32}$$

This estimate is valid for all m_σ, not just $m_\sigma \geqslant 0$. All relevant multi-indices can be obtained in a recursion similar to the above. The line "if $\partial^\alpha(p)$ is monotone" only needs to be replaced by "if $\partial^\alpha(p) = 0$."

With these remarks it is now clear that Algorithm 4, applied to polynomials with the estimates (28) or (32), can be implemented as a *black box* algorithm: the only input required is the coefficients of the polynomials and an initial interval. An implementation in JAVA is a current project; see [3].

7. SUMMARY

An application of Algorithm 4 typically consists of three steps:

1. Given the problem $\min_{x \in \Lambda} f(x)$, construct a g such that $f \prec_q g$. The results in Section 3 are used in this step.

2. Implement the estimates (18) or (20) for the given q. Note that for $q > 1$ many partial derivatives are involved, so we have to construct a MAPLE script that actually implements these estimates (say, in JAVA), once f and g are given.

3. Run Algorithm 4 based on the estimate constructed in step 2.

4. A typical feature of Algorithm 4 is that each global minimal point causes the generation of several intervals, and therefore in a final step we have to sort out which intervals represent the same global minimal point. We call two intervals in Γ_ℓ close iff their midpoints m_1 and m_2 satisfy an inequality $|m_1 - m_2| \leqslant C 2^{-\ell} r$ where $r = r_A$ is the radius of the initial interval. Ideally, $C = 2$; however, a more practical choice is some constant $C > 2$. This notion of closeness defines *connected components* in the set Γ_ℓ. Lemma 22 implies the existence of a $C \geqslant 2$ such that for sufficiently large ℓ each global minimal point is represented by exactly one connected component of intervals. We say that the algorithm has isolated all global minimal points (for such ℓ). It is not difficult to write a program that generates such connected components.

Note that for polynomials, items 1 and 2 can be automated and incorporated directly into an implementation of Algorithm 4 as indicated in Section 6.

ACKNOWLEDGMENT

REFERENCES

1. A. Arazyan and R. B. Kearfott, Taylor series models in deterministic global optimization, *in* "Proceedings of AD 2000, the Third International Conference and Workshop on Automatic Differentiation, June 19–23, 2000, Nice, France," SIAM, Philadelphia, in press.
2. T. Csendes and D. Ratz, Subdivision direction selection in internal methods for global optimization, *SIAM J. Numer. Anal.* **34** (1997), 922–938.
3. M. Erdmann, "On the Implementation and Analysis of Cellular Exclusion Algorithms," Ph.D. thesis, Colorado State University, 2001, in preparation.
4. K. Georg, Improving the efficiency of exclusion algorithms, manuscript submitted for publication.
5. K. B. Kearfott, Abstract generalized bisection and a costbound, *Math. Comput.* **49** (1987), 187–202.
6. R. B. Kearfott, "Rigorous Global Search: Continuous Problems," Kluwer Academic, Dordrecht, 1996.
7. R. B. Kearfott, Empirical evaluation of innovations in internal branch and bound algorithms for nonlinear algebraic systems, *SIAM J. Sci. Comput.* **18** (1997), 574–594.
8. R. B. Kearfott and K. Du, The cluster problem in multivariate global optimization, *J. Global Optim.* **5** (1994), 253–265.
9. R. E. Moore, "Methods and Applications of Interval Analysis," SIAM, Philadelphia, 1979.

10. H. Ratschek and J. Rokne, "New Computer Methods for Global Optimization," Wiley, New York, 1988.
11. Z.-B. Xu, J.-S. Zhang, and Y.-W. Leung, A general CDC formulation for specializing the cell exclusion algorithms of finding all zeros of vector functions, *Appl. Math. Comput.* **86** (1997), 235–259.
12. Z.-B. Xu, J.-S. Zhang, and W. Wang, A cell exclusion algorithm for determining all the solutions of a nonlinear system of equations, *Appl. Math. Comput.* **80** (1996), 181–208.