J Glob Optim (2007) 37:27–45
DOI 10.1007/s10898-006-9034-z

ORIGINAL ARTICLE

Theoretical convergence analysis of a general division–deletion algorithm for solving global search problems

X. Yang · M. Sun

Received: 8 September 2005 / Accepted: 17 April 2006 /
Published online: 6 July 2006

Abstract Presented in this paper is the prototype of a very general algorithm referred to as Division – Deletion Algorithm (DDA) for solving the most general global search problem. Various necessary conditions, sufficient conditions, and necessary and sufficient conditions for the convergence of the algorithm are proposed and analyzed. As an example of its application, we demonstrate that the convergence of a standard Hansen's interval algorithm for unconstrained global optimization simply follows from this general theory.

Keywords General global search problem · Algorithm prototype · Optimization · Convergence analysis

1 Introduction

Developing convergent searching algorithms for locating the globally optimal value (f^*) of an objective function $f(x)$ and at least one global optimizer over a bounded domain X in $\mathbf{R}^d$, possibly subject to other equality and inequality constraints, is a challenging mathematical problem. The problem can be stated as

$$\begin{aligned}&\text{minimize } f(x),\\&\text{subject to } g(x) \leq 0, \quad h(x) = 0, \quad x \in X.\end{aligned}$$

Due to its enormous theoretical and practical importance, a lot of efforts by researchers from a wide range of disciplines have been devoted to solving this problem.

X. Yang (✉) · M. Sun
Department of Mathematics, The University of Alabama,
Tuscaloosa, AL 35487-0350,
USA
e-mail: yang009@bama.ua.edu

M. Sun
e-mail: msun@gp.as.ua.edu

Classical mathematical theory of smooth local optimization and local search algorithms are well known. However, the global problem presents several difficult issues. One such fundamentally difficult issue is the fact that there is no single verifiable sufficient condition for a globally optimal solution unless it is a very special case. Thus either a global behavior of the functions involved is taken into account or the entire search domain is examined by global search algorithms. Two large categories of global search algorithms are available: deterministic and stochastic. Deterministic algorithms are generally based on the idea of division and bounding, which include branch and bound methods (cf. Lawler and Wood 1966; Horst 1976; Horst and Tuy 1990), interval methods (cf. Moore 1979; Alefeld and Herzberger 1983; Ratscheck and Rokne 1988; Neumaier 1990; Hansen 1992; Kearfott 1996; Sun and Johnson 2005), and cell exclusion methods (cf. Xu, et al. 1997). Stochastic algorithms include a probabilistic movement mechanism that makes it probabilistically possible to escape from locally optimal solutions (cf. Kirkpatrick et al. 1983; Goldberg 1989; Sun 2002). Of course, our list of references on those methods is far from complete. Some deterministic algorithms are capable of locating all the global solutions at an expense of more CPU time and memory. Stochastic methods do not guarantee convergence to a global solution when a particular run ends. But they are usually easier to implement and quicker to reach an approximate solution.

This paper introduces a very general prototype of search algorithms for locating all the solutions to the most general global problem. The global problem would include the global optimization problem stated earlier in this section, among many others. We are not concerned with any particular implementation of the algorithm and any particular formulation of the global problem. However, we will be focusing on identifying conditions that are characteristically related to the theoretical convergence of the algorithm at the basic level of abstraction. We propose four types of characteristic conditions that are proved to be closely related to the convergence of the general algorithm. Various necessary conditions, sufficient conditions, and necessary and sufficient conditions for the convergence of the general algorithm are introduced and analyzed. As an example, of its application and connection to existing literature, we demonstrate near the end of the paper that the convergence of a standard Hansen's interval algorithm for the unconstrained global optimization (cf. Hansen 1980; Ratscheck and Rokne 1988) readily follows from our general theory. Thus, the paper contributes to existing literature on global search (including global optimization) through the basic level of abstraction.

The rest of the paper is organized as follows. In Sect. 2, the formulation of the most general global search problem is defined along with a list of some specific types of global problems. In Sect. 3, we introduce a general prototype of search algorithms referred to as Division–Deletion Algorithm (DDA) for solving the global problem. Four characteristic ingredients of DDA are also discussed in that section along with its relationship with few well-known existing methods. In Sect. 4, four categories of conditions for the convergence of DDA are given. Relationships among those conditions are also discussed in that section. In Sect. 5, sufficient conditions and necessary conditions for the convergence of DDA are established. From those results, we derive two sets of necessary and sufficient conditions of convergence. To conclude Sect. 5, we present two counterexamples for some of the necessary conditions and sufficient conditions presented earlier in that section. In Sect. 6, we prove that a standard Hansen's interval algorithm can be treated as an example of our DDA and demonstrate that it satisfies a set of sufficient conditions listed in Sect. 5. Finally in Sect. 7, we summarize our results and briefly discuss future work about DDA.

2 Formulation of general global search problem

Any problem seeking for all points contained in a given set X and satisfying a given property P^* is called a global search problem. Its solution set is denoted by X^*, i.e.,

$$X^* = \{x \in X | x \text{ satisfies the property } P^*\}.$$

Each point of X^* is called a global solution of the problem. This formulation could be made more precise in terms of the logic terminology. Let $P(x)$ be a predicate defined over the domain X . Then finding the truth set of the predicate $P(x)$ over X is the global problem. Thus, this is perhaps the most general global problem stated up to date. The search problem is global since the property P^* and the predicate $P(x)$ may depend on the whole domain. The following problems are typical examples of such a global problem for a given set X and given functions $f(x), g(x)$, and $h(x)$ over X.

(P_1) Minimize $f(x)$ over X.
(P_2) Minimize $f(x)$ over X, subject to $g(x) \leq 0$ and $h(x) = 0$.
(P_3) Solve $h(x) = 0$ over X.
(P_4) Solve $g(x) \leq 0$ over X.
(P_5) Solve $g(x) \leq 0$ and $h(x) = 0$ over X.
(P_6) Solve $g(x) < 0$ over X.

There are lots of other kinds of global search problems not as commonly quoted in research literatures as those listed above. Here are a couple of samples that are of some practical interest.

(P_7) Search for discontinuities of $f(x)$ over X.
(P_8) Find all values of the parameter α such that the differential equation $u_{xx} + \alpha u = 0$ ($x \in X \subset \mathbf{R}^1$) with the boundary value condition $u|_{\partial X} = 1$ has a unique solution.

In principle, we do not need to impose any restrictions on the set X. However, in view of the practical applications that we have in mind, X is considered to be a subset of a finite dimensional Euclidean space $\mathbf{R}^d (1 \leq d < \infty)$. We are going to introduce a general search algorithm for solving the global problem, and look for sufficient conditions and/or necessary conditions for the convergence of the algorithm.

3 Division–deletion algorithm

3.1 Division–deletion algorithm

In this section, we propose a general prototype of a class of global search algorithms, DDA, or simply DD, for solving the general global search problem. The algorithm follows the basic division-bounding principle as done in several existing algorithms for the case of global optimization. However, we are not concerned with any specific implementation of the algorithm.

Division–Deletion Algorithm:

Step 1. Set $Y = X$ and initialize the list $L = \{Y\}$. Set iteration counter n = 1.
Step 2. Subdivide Y into subsets $V_1, \ldots, V_s$ such that $Y = \cup_{j=1}^{s} V_j$.
Step 3. Check a deletion condition on the subsets $V_1, \ldots, V_s$. Discard those of $V_1, \ldots, V_s$ satisfying the deletion condition. Place the others into the list L. Then remove Y from L.
Step 4. If a termination criterion holds, go to Step 7.
Step 5. Select a conditional set Y (cf. Remark 3.1 below) from L. If such a selection is impossible, then go to Step 7.
Step 6. Set n equal to $n + 1$, and go to Step 2.
Step 7. The algorithm ends.

Remark 3.1 A set in the list L will be referred to as a conditional set if it does not satisfy the deletion condition. However, it is optional to determine whether a conditional set is completely contained in the solution set or not.

Remark 3.2 Let M be a conditional set in the list L. Then one of the following two cases will occur.

Case 1. $M \cap X^* = M$, i.e. $M \subseteq X^*$.
Case 2. $M \cap X^* = \phi$ or $M - X^* \neq \phi$ and $M - X^* \neq M$.

3.2 Additional notation

- $L_n = L$ at iteration n (see Step 6) of DDA.
- $U_n =$ the union of all the sets of L_n.
- $d(M) = \sup_{u,v \in M}\{\|u - v\|_2\}$ is called the diameter of M, where $\|u - v\|_2$ is the 2-norm.
- $\mu(M) = \inf_{M \subseteq \underset{k\geq 1}{\cup} I_k} \sum_{k\geq 1} \mu(I_k)$ is called the outer measure of $M \subseteq \mathbf{R}^d$, where

$$I_k = (a_1, b_1) \times (a_2, b_2) \times \cdots \times (a_d, b_d), \mu(I_k) = \prod_{i=1}^{d}(b_i - a_i)$$

and the collection $\{I_k\}_{k\geq 1}$ represents any countable covering of M (cf. Rudin 1964). If $d = 1, \mu(M)$ is denoted by $\mu_0(M)$ (cf. Royden 1968). If M is a Lebesgue measurable set, $\mu(M)$ is also the Lebesgue measure of M.

- $P_i(x) = x_i$ is called the projection of $x = (x_1, \ldots, x_i, \ldots, x_d)^T \in \mathbf{R}^d$ onto the ith direction of $\mathbf{R}^d$.
- $P_i(M) = \{P_i(x) | x \in M\}$ is called the projection of $M \subseteq \mathbf{R}^d$ onto the ith direction of $\mathbf{R}^d$.
- $\mu_i(M) = \mu_0(P_i(M))$.
- For any nonempty subsets A and B of $\mathbf{R}^d$,

$$\begin{aligned} d_0(x, B) &= \inf_{b\in B} \|x - b\|_2, \\ d_0(A, B) &= \sup_{a\in A}\{d_0(a, B)\}, \\ d(A, B) &= \max\{d_0(A, B), d_0(B, A)\}, \end{aligned}$$

- $\lim_{n\to\infty} A_n = B$ or $A_n \to B$ as $n \to \infty$ if $\lim_{n\to\infty} d(A_n, B) = 0$, where $\{A_n\}_{n=1}^{\infty}$ is a sequence of nonempty sets in $\mathbf{R}^d$ and $B \subseteq \mathbf{R}^d$ is also not empty (cf. Ratscheck and Rokne 1988).

Remark 3.3 If $\{A_n\}_{n=1}^{\infty}$ is a decreasing sequence of sets (i.e. $A_{n+1} \subseteq A_n$) with some $A_m = \phi$, then we still say that $\lim_{n\to\infty} A_n = \phi$.

Convergence (C): The DDA is said to be convergent if the decreasing sequence $\{U_n\}_{n=1}^{\infty}$ of unions of the sets generated but not discarded by DDA converges to X^*, that is, $\lim_{n\to\infty} U_n = X^*$, or equivalently $X^* = \cap_{n=1}^{\infty} U_n$.

Remark 3.4 This definition of convergence is stronger than usual. Normally, a global search algorithm is considered as convergent if it can identify a global solution. The same remark applies to the definition of general global search problem in Sect. 2.

3.3 Essential ingredients of DDA

3.3.1 Division

Our newly proposed DDA tries to search through the conditional sets in the sequentially generated lists for all the global solutions instead of searching the original domain X uniformly. Thus, during the process of DDA, a conditional set in the current list will be divided into subsets as new candidates for forming a subsequent list. For example, in a standard Hansen's algorithm mentioned earlier, the bisection was used. In a multi-splitting algorithm (cf. Csallner et al. 2000), a multisection strategy was considered. Since we are dealing with very general global problem, repeated division of the original domain is essential for identifying any global solution by this deterministic algorithm.

3.3.2 Deletion condition

Division alone would create more and more conditional sets. It leads to a serious computational burden. Thus, our DDA uses a deletion (discarding) condition as another key ingredient. Any conditional set that satisfies the deletion condition will be discarded from the list so that it will never be considered by DDA again. Thus, deletion conditions make it possible that the total search region is getting smaller and smaller until all the global solutions are found. Deletion conditions can be set up based on specific global search problems. For example, the midpoint test was used in the Hansen's algorithm.

The deletion step of DDA has been stated for newly generated subsets. It could be applied to other sets in the current list if it is beneficial to do so. Such modification does not effect the theoretical results presented in this paper. In fact, many other improvement strategies could be incorporated without altering the main convergence results.

3.3.3 Selection

Usually, the conditional set in the current list is not unique. Thus, selection of a conditional set in Step 5 of DDA becomes an issue. For convenience, an ordering of the sets in a list can be introduced. The leading or first set in each list will be selected for the subsequent division. In the Hansen's algorithm presented in Sect. 6.1, the order of sets in a list was determined by the age of the sets. However, in Moore–Skelboe algorithm (cf. Ratscheck and Rokne 1988), the set with the smallest lower bound of its inclusion function was selected. In another algorithm (cf. Csendes 2001), a different subset selection criterion was proposed. However, the selection cannot be arbitrary if the algorithm is going to converge. We will introduce conditions that are affected by the selection strategy. Thus selection is considered as another key ingredient of DDA.

3.3.4 Termination criterion

For practical reasons, we might hope that the global solution set would be found with some desired accuracy. The algorithm needs to be stopped when the solution set can be identified so that unnecessary steps could be avoided. If the size of a set is measured by its diameter or some other measure, then we may stop the algorithm when the size of each conditional set gets small enough so that the global solution set can be estimated effectively.

To end this section, we take another brief look at our DDA in the light of well-established branch and bound methods and interval algorithms for the case of global optimization. Although they share some similarity in using the division-bounding principle as pointed out earlier, they display significant differences. A standard prototype branch and bound method (see Horst and Tuy 1990) uses a consistent bounding operation and does not find all the global solutions. Interval algorithms could be considered as branch and bound methods that use interval arithmetic. Most interval algorithms use inclusion functions for bounding and they capture all the global solutions. The DDA is a lot more general than the others. It does not use any specific objects such as intervals, inclusion functions, and bounding functions. In fact, it does not explicitly use any function at all. It is designed not just for solving global optimization problems. Thus, it provides a more basic level of abstraction. Concrete implementation procedures must be designed for its steps before any global problem can be solved. Section 6 provides an existing example of implementation. In fact, many of existing branch and bound procedures also fit into the prototype of DDA. Implementations of DDA would involve more specific objects. However, such objects are not limited to functions and intervals. Despite the generality of DDA and the global problem itself, we are still able to offer fairly comprehensive convergence analysis by providing various versions of necessary conditions, sufficient conditions, and necessary and sufficient conditions for the convergence (to all the global solutions) of DDA. Sufficient conditions for convergence of algorithms can be found fairly frequently in the optimization literature. Necessary conditions for convergence are rarely discussed. Necessary and sufficient conditions for convergence can hardly be found in any nontrivial situations. But, we will present all those conditions in the next two sections.

4 List of conditions and their properties

In this section, we introduce various versions of four categories of conditions that are found to be fundamentally associated with the convergence. Their properties and relationships are also discussed. They will further be classified as necessary and/or sufficient conditions in the next section.

Regarding the conditions relevant to the convergence, first we impose these underlying assumptions for the rest of the paper (unless we state otherwise), as follows:

(1) The search domain X is compact.
(2) Each conditional set is connected.
(3) Each subdivision of a set Y creates a partition $\{V_1, V_2, \ldots, V_s\}$ of Y in the usual sense. That is, the V_i's are closed, $Y = \cup_{j=1}^{s} V_j$, and $V_i \cap V_j = \partial V_i \cap \partial V_j (i, j \in \{1, 2, \ldots, s\},\ i \neq j)$, where ∂V_i represents the boundary of V_i relative to Y. It follows that each conditional set is also compact.
(4) The algorithm does not reach a desired stop in a finite number of steps.

The first category of conditions regards the way of measuring the closeness of conditional sets to the solution set when they are intersected. It is affected by the selection and subdivision strategies used in the algorithm. The second category is about the deletion condition that indicates that there is no global solution lost during the search process. The third category deals with those conditional sets not intersected with the solution set. They should eventually be deleted by the algorithm. The fourth category is related to the properties of the solution set of the original global problem. We will demonstrate in the next section that the desired convergence of DDA cannot afford to lose any one category of those conditions, and does not require any more categories either. Those conditions (possibly with multiple versions) are listed below in detail.

Condition 1

Let $\{M_n\}_{n=1}^{\infty}$ be any sequence of conditional sets generated by DDA with $M_n \cap X^* \neq \phi$ for all $n \geq 1$. We implicitly assume $M_n \in L_n$ for all $n \geq 1$.

Cla $\lim_{n\to\infty} d(M_n) = 0$.
Clb $\lim_{n\to\infty} d(M_n - X^*) = 0$.
Clc $\lim_{n\to\infty} \mu_i(M_n - X^*) = 0\ (1 \leq i \leq d)$.
Cld $\lim_{n\to\infty} d_0(M_n, X^*) = 0$.
Cle $\lim_{n\to\infty} \mu(M_n) = 0$.
Clf $\lim_{n\to\infty} \mu(M_n - X^*) = 0$.
Clg $\lim_{n\to\infty} \mu_i(M_n) = 0\ (1 \leq i \leq d)$.

Condition 2a (C2a)

All the global solutions are contained in the union of the sets of each list. In other words, no global solution gets lost during the course of DDA, that is, $X^* \subset U_n$ for all $n \geq 1$.

Condition 3a (C3a)

Any conditional set not containing any global solution will be completely deleted after a finite number of steps. Equivalently, if V' is a conditional set of some list $L_{n'}$ generated by DDA with $V' \cap X^* = \phi$, then there is an integer $n'' > n'$ such that $V' \cap U_n = \phi$ whenever $n > n''$.

Condition 4

C4a The solution set X^* is closed.
C4b The solution set X^* is closed with $\mu_i(X^*) = 0\ (1 \leq i \leq d)$.
C4c The solution set X^* is closed with $\mu(X^*) = 0$.

When $X^* = \phi$, any nonempty set in the list would be a conditional set at least in theory. In this case, C3a alone would be necessary and sufficient for the convergence of DDA. In other words, DDA can still detect the correct X^*. Thus, we assume $X^* \neq \phi$ for the rest of the paper.

In the remainder of this section, we explore various relationships among those conditions. Some straightforward proofs will be skipped in our presentation below.

Property 4.1 *If $\{M_n\}_{n=1}^{\infty}$ is a decreasing sequence of subsets of $\mathbf{R}^d$, so is $\{M_n - X^*\}_{n=1}^{\infty}$.*

Property 4.2 *If $M \subseteq \mathbf{R}^d$ satisfies $\mu_i(M) = 0 \quad (1 \leq i \leq d)$, then $\mu(M) = 0$.*

Property 4.3 *For any conditional set M,*

(a) $P_i(M)$ is compact for $1 \leq i \leq d$;
(b) $\mu(M) = \mu(M - X^*)$ if $\mu(X^*) = 0$;
(c) $\mu_i(M) = \mu_i(M - X^*)$ if $\mu_i(X^*) = 0$ for $1 \leq i \leq d$.

Property 4.4 *Suppose $A, C, B \subseteq \mathbf{R}^d$ are not empty. The following statements are true.*

(a) $d_0(A \cup C, B) \leq d_0(A, B) + d_0(C, B)$.
(b) $d_0(C, B) \leq d_0(C, A)$ if $A \subseteq B$.
(c) $d_0(A, C) \leq d_0(B, C)$ if $A \subseteq B$.
(d) $d_0(A, B) = 0$ if $A \subseteq B$.

Property 4.5 *For any conditional set M,*

(a) $d_0(M, X^*) = 0$ if $M \cap X^* = M$;
(b) $d_0(M, X^*) = d_0(M - X^*, X^*)$ if $M \cap X^* \neq M$;
(c) $d_0(M, X^*) \leq \max_{1 \leq i \leq d} \mu_i(M - X^*)\sqrt{d}$ if $M \cap X^* \neq \phi$

Proposition 4.1

(a) C1e $\Leftrightarrow$ Clf, if C4c *is satisfied.*
(b) C1c $\Leftrightarrow$ Clg, if C4b *is satisfied.*
(c) C4b $\rightarrow$ C4c.

Proof It follows from Propertys 4.2 and 4.3(b)–(c). □

Proposition 4.2

(a) C1a $\Leftrightarrow$ C1g;
(b) C1a $\rightarrow$ C1b;
(c) C1b $\rightarrow$ C1c;
(d) C1c $\rightarrow$ C1d;
(e) C1c $\rightarrow$ C1f;
(f) C1e $\rightarrow$ C1f;
(g) C1g $\rightarrow$ C1e.

Proof Suppose that $\{M_n\}_{n=1}^{\infty}$ is any sequence of conditional sets with $M_n \cap X^* \neq \phi$ for all $n \geq 1$.

(a) The equivalence is established in two steps.
(a1) Suppose $\lim_{n\to\infty} d(M_n) = 0$. Since $0 \leq \mu_i(M_n) \leq d(M_n) \to 0$ as $n \to \infty$, $\lim_{n\to\infty} \mu_i(M_n) = 0 (1 \leq i \leq d)$.

(a2) Suppose $\lim_{n\to\infty} \mu_i(M_n) = 0 (1 \le i \le d)$. Since M_n is connected for all $n \ge 1$, $P_i(M_n)$ is a closed interval on the ith axis $(1 \le i \le d)$. Write

$$I_{M_n} = P_1(M) \times P_2(M) \times \cdots \times P_d(M).$$

Then,$M_n \subseteq I_{M_n}$ and $\mu_i(I_{M_n}) = \mu_i(M_n) \to 0 (1 \le i \le d)$ as $n \to \infty$. Clearly, $\lim_{n\to\infty} d(I_{M_n}) = 0$.

(b) or (f) It is easy to see.
(c) It is similar to the proof in (a1).
(d) By Property 4.5(c),

$$d_0(M_n, X^*) \le \max_{1\le i\le d} \mu_i(M_n - X^*)\sqrt{d}.$$

Thus, we have $\lim_{n\to\infty} d_0(M_n, X^*) = 0$ when $\lim_{n\to\infty} \mu_i(M_n - X^*) = 0 \quad (1 \le i \le d)$.

(e) or (g) It is similar to the proof in (a2). □

Corollary 4.1 C1a → C1e.

According to Propositions 4.1(b) and 4.2(a), we get the following result.

Corollary 4.2 C1a ⇔ Clc, if C4b *is satisfied.*

Remark 4.1 C1a ⇔ C1g ⇔ C1c, if C4b is satisfied.

5 Necessary and/or sufficient conditions for the convergence of DDA

This section deals with classification of the conditions introduced in the previous section as necessary and/or sufficient conditions for the convergence of DDA. To prove such claims, we first state some preliminary results with proofs omitted.

Property 5.1 *Suppose that* $\{M_n\}_{n=1}^{\infty}$ *is a decreasing sequence of nonempty compact subsets of* $\mathbf{R}^d$. *Then*

(a) $\cap_{n=1}^{\infty} M_n$ is compact;
(b) $\cap_{n=1}^{\infty} M_n \ne \phi$;
(c) $\lim_{n\to\infty} M_n = \cap_{n=1}^{\infty} M_n$;
(d) $\cap_{n=1}^{\infty} P_i(M_n) = P_i(\cap_{n=1}^{\infty})$ or equivalently $\lim_{n\to\infty} P_i(M_n) = P_i(\lim_{n\to\infty} M_n)$ for $1 \le i \le d$.

Remark 5.1 The results in Property 5.1(c) and (d) remain valid without the assumption that M_n is nonempty for all $n \ge 1$.

5.1 Sufficient conditions

When a new convergent algorithm is proposed, sufficient conditions of the convergence are usually stated and proved. Thus, we discuss sufficient conditions first.

Theorem 5.1 *Under four conditions* C1d, C2a, C3a, and C4a, *the decreasing sequence* $\{U_n\}_{n=1}^{\infty}$ *generated by* DDA *converges to* X^*, that is, $X^* = \cap_{n=1}^{\infty} U_n$.

Proof Clearly, $X^* \subseteq \cap_{n=1}^{\infty} U_n$ since C2a is satisfied. Next, we will show that

$$X^* \supseteq \bigcap_{n=1}^{\infty} U_n.$$

Suppose $X^* \supseteq \cap_{n=1}^{\infty} U_n$ is not true. Then there is a point $x' \in U_n$ for all $n \geq 1$, but $x' \notin X^*$. Thus, for any $n \geq 1$, there is a conditional set, denoted by M_n, of the list L_n generated by DDA that contains x' and satisfies $M_n \subseteq M_{n-1}$, where $M_0 = X$. We also claim that $M_n \cap X^* \neq \phi$ for all $n \geq 1$. In fact, if $M_{n'} \cap X^* = \phi$ for some $n' \geq 1$, then $M_{n'}$ would be completely deleted after a finite number of steps according to C3a and $\{M_n\}_{n\geq 1}$ is only a finite sequence. This causes a contradiction. Thus C1d can be applied to this sequence $\{M_n\}_{n\geq 1}$

$$\lim_{n\to\infty} d_0(M_n, X^*) = 0.$$

On the other hand, since X^* is closed (i.e. C4a),

$$d_0(x', X^*) > 0.$$

Now we consider

$$d_0(M_n, X^*) = \sup_{x\in W_n} d_0(x, X^*) \geq d_0(x', X^*) > 0 \quad \text{for all} \quad n \geq 1.$$

Thus,

$$\lim_{n\to\infty} d_0(M_n, X^*) \geq d_0(x', X^*) > 0,$$

which is contradictive to C1d. Therefore, $x' \in X^*$ and the desired result follows. □

Based on Theorem 5.1 and Proposition 4.2, we get the following results.

Corollary 5.1 *Any of the following three sets of conditions would be sufficient for the convergence of* DAA.

(a) C1a, C2a, C3a, and C4a.
(b) C1b, C2a, C3a, and C4a.
(c) C1c, C2a, C3a, and C4a.

5.2 Necessary conditions

Theorem 5.2 *Suppose that the decreasing sequence $\{U_n\}_{n=1}^{\infty}$ generated by* DDA *converges to X^* and $\{M_n\}_{n=1}^{\infty}$ is any sequence of conditional sets with $M_n \cap X^* \neq 0$ for all $n \geq 1$. Then the following statements are true as follows:*

(a) *All the global solutions are contained in the union of the sets in each list* (C2a).
(b) *Each conditional set that does not contain any global solution will be completely deleted after a finite number of steps* (C3a).
(c) *The solution set X^* is closed* (C4a).
(d) $\lim_{n\to\infty} \mu(M_n - X^*) = 0$ (Clf).
(e) $\lim_{n\to\infty} \mu_i(M_n) = 0, 1 \leq i \leq d$ (C1g) if C4b *is satisfied.*
(f) $\lim_{n\to\infty} d_0(M_n, X^*) = 0$ Cld.

Proof (a) Suppose not. Then there is at least one global solution that is not in the union of all sets of some list. Thus, $\cap_{n=1}^{\infty} U_n \neq X^*$, contradicting to the assumed convergence. Therefore, C2a is necessary.
(b) Suppose not. Then there is a sequence of conditional sets $\{W_n\}_{n=1}^{\infty}$ generated by DDA with $W_{n+1} \subseteq W_n$ such that $W_n \cap X^* = \phi$ but $\cap_{n=1}^{\infty} W_n \neq \phi$ by Property 5.1(b). Note that $(\cap_{n=1}^{\infty} W_n) \cap X^* = \phi$ and $\cap_{n=1}^{\infty} W_n \subseteq \cap_{n=1}^{\infty} U_n$. Thus, we have $X^* \neq \cap_{n=1}^{\infty} U_n$, contradicting to the assumed convergence $X^* = \cap_{n=1}^{\infty} U_n$. Therefore, C3a is necessary as well.
(c) Suppose not again. Then there is a sequence, $\{x_k^*\}_{k=1}^{\infty}$, of global solutions such that $\lim_{k\to\infty} x_k^* = x' \notin X^*$. Note that $x' \in U_n$ since U_n is closed for all $n \geq 1$. Hence $x' \in \cap_{n=1}^{\infty} U_n$. Thus, to $\cap_{n=1}^{\infty} U_n \neq X^*$, contradicting to $\cap_{n=1}^{\infty} U_n = X^*$ again. Therefore, X^* must be closed.
(d) It is easy to see that $\{U_n - X^*\}_{n=1}^{\infty}$ is decreasing according to Property 4.1. Thus, $\{(\mu U_n - X^*)\}$ is also decreasing, and bounded from below by 0. Thus, $\lim_{n\to\infty} \mu(U_n - X^*)$ exists. On the other hand, since X is compact,

$$\mu(U_n - X^*) \leq \mu(X) < \infty \quad \text{for all } n \geq 1.$$

Therefore,

$$\lim_{n\to\infty} \mu(U_n - X^*) = \mu\left(\bigcap_{n=1}^{\infty}(U_n - X^*)\right) \text{ (cf. Royden 1968).}$$

If $\lim_{n\to\infty} \mu(U_n - X^*) \neq 0$, that is, $\mu\left(\bigcap_{n=1}^{\infty}(U_n - X^*)\right) \neq 0$,then

$$\bigcap_{n=1}^{\infty}(U_n - X^*) \neq \phi.$$

Take $x' \in \cap_{n=1}^{\infty}(U_n - X^*)$. Thus, $x' \in U_n - X^*$ for all $n \geq 1$. Due to the convergence of DDA,

$$\bigcap_{n=1}^{\infty}(U_n - X^*) \subseteq \bigcap_{n=1}^{\infty} U_n = X^*.$$

Thus, $x' \in X^*$, giving us a contradiction. Therefore,

$$0 \leq \lim_{n\to\infty} \mu(M_n - X^*) \leq \lim_{n\to\infty} \mu(U_n - X^*) = \mu\left(\bigcap_{n=1}^{\infty}(U_n - X^*)\right) = 0.$$

(e) (e1) By the convergence of DDA,

$$\bigcap_{n=1}^{\infty} U_n = X^* \text{ and } 0 \leq \mu_i\left(\bigcap_{n=1}^{\infty} U_n\right) = \mu_i(X^*) = 0 \quad (1 \leq i \leq d).$$

Thus, $\mu_0\left(P_i\left(\cap_{n=1}^{\infty} U_n\right)\right) = 0$. By Property 5.1(d), $\cap_{n=1}^{\infty} P_i(U_n) = P_i\left(\cap_{n=1}^{\infty} U_n\right)$. Thus,

$$\mu_0\left(\bigcap_{n=1}^{\infty} P_i(U_n)\right) = \mu_0\left(P_i\left(\bigcap_{n=1}^{\infty} U_n\right)\right) = 0 \quad (1 \leq i \leq d).$$

(e2) Due to compactness of X, by Property 4.3(a), $P_i(X)$ is also compact with

$$\mu_i(U_n) \leq \mu_i(X) < \infty \quad (1 \leq i \leq d).$$

Since $\{(U_n)\}_{n=1}^{\infty}$ is a deceasing sequence of compact sets, $\{P_i(U_n)\}_{n=1}^{\infty}$ is also a decreasing sequence of compact sets. Thus (cf. Royden 1968),

$$0 \le \overline{\lim_{n\to\infty}} \mu_i(M_n) \le \lim_{n\to\infty} \mu_i(U_n) = \lim_{n\to\infty} \mu_0(P_i(U_n)) = \mu_0\left(\bigcap_{n=1}^{\infty} P_i(U_n)\right) \quad \text{for } 1 \le i \le d.$$

Therefore, combining with (e1), $\lim_{n\to\infty} \mu_i(M_n) = 0 \quad (1 \le i \le d)$.
(f) Since $\lim_{n\to\infty} U_n = X^*$, $\lim_{n\to\infty} d(U_n, X^*) = 0$. Thus,

$$\lim_{n\to\infty} \max\{d_0(U_n, X^*), d_0(X^*, U_n)\} = 0.$$

According to (a), C2a is satisfied, i.e., $X^* \subseteq U_n$ for all $n \ge 1$. Thus, by Property 4.4(d),

$$d_0(X^*, U_n) = 0, \quad \text{for all } n \ge 1.$$

Therefore, $\lim_{n\to\infty} d_0(U_n, X^*) = 0$.

Note that $M_n \subseteq U_n$ for all $n \ge 1$. Thus, according to Property 4.4(c),

$$0 \le d_0(M_n, X^*) \le d_0(U_n, X^*) \text{ for all } n \ge 1.$$

Hence, $\lim_{n\to\infty} d_0(M_n, X^*) = 0$. □

According to Theorems 5.2(d), 5.2 (e), Proposition 4.1(a), and Remark 4.1, we have the following results.

Corollary 5.2 *Suppose that the decreasing sequence* $\{U_n\}_{n=1}^{\infty}$ *generated by* DDA *converges to* X^* *and* $\{M_n\}_{n=1}^{\infty}$ *is any sequence of conditional sets with* $M_n \cap X^* \neq \phi$ *for all* $n \ge 1$. *Then*

(a) $\lim_{n\to\infty} \mu(M_n) = 0$ (Cle) *if* C4c *is satisfied;*
(b) $\lim_{n\to\infty} \mu_i(M_n - X^*) = 0, \quad 1 \le i \le d$ (C1c) *if* C4b is satisfied;
(c) $\lim_{n\to\infty} d(M_n) = 0$ (C1a) *if* C4b *is satisfied.*

We have identified nine necessary conditions along with the four sets of sufficient conditions presented earlier for the convergence of DDA. Putting these together, we are able to state four sets of necessary and sufficient conditions below.

5.3 Necessary and sufficient conditions

According to Theorems 5.1 and 5.2(f), we have the next result.

Theorem 5.3 *Conditions* C1d, C2a, C3a, and C4a *together are necessary and sufficient for the convergence of* DDA.

According to Corollarys 5.1(a), 5.2(c) and Theorem 5.2(e), we get the following three sets of necessary and sufficient conditions for the convergence of DDA.

Theorem 5.4 If $\mu_i(X^*) = 0$ $(1 \le i \le d)$, *then conditions* C2a, C3a, and C4a, *together with any of* C1a, C1g, and C1c, *are necessary and sufficient for the convergence of* DDA.

The necessary and sufficient conditions of convergence are the most important results of the paper. However, we have spread out their proofs in series of properties, propositions, theorems, and Corollaries in the previous part of the paper. Theorem 5.3 applies to the general case, while Theorem 5.4 can be applied to some important cases in which there are only at most a countably many global solutions.

5.4 Counterexamples

We now present two counterexamples to demonstrate that some of the necessary conditions are not sufficient while some sufficient conditions are not necessary. The examples use interval arithmetic. Interval related notations are described in Section 6 and references therein.

Example 5.1

$$(p_4^1) \text{ Solve } x \le 0 \text{ and } -1 \le y \le 1 \text{ over } [-1,1] \times [-1,1].$$

It is very easy to get the solution set $X^* = [-1,0] \times [-1,1]$. We can also construct a simple convergent interval algorithm, Algorithm (A_4^1) below, for solving Problem (P_4^1). We first choose

$$F(T) = T \text{ for } T \in I([-1,1])$$

as an inclusion function of $f(x) = x$ (for $x \in [-1,1]$). Then we construct this interval algorithm.

Algorithm $\left(A_4^1\right)$

Step 1 Set $Y = [-1,1], y = lbF(Y)$, and $\bar{y} = ubF(Y)$. Initialize the list

$$L = \{(Y \times [-1,1], y, \bar{y})\}.$$

Step 2 Bisect $Y \times [-1,1]$ into two interval boxes $V_1 \times [-1,1]$ and $V_2 \times [-1,1]$ along the direction perpendicular to Y at the point that is $|w(Y) - 1|/2$ away from the right endpoint of Y such that $Y = V_1 \cup V_2$ and $V_1 \le V_2$. Form two new triplets $(V_1 \times [-1,1], v_1, \bar{v}_1)$ and $(V_2 \times [-1,1]), v_2, \bar{v}_2)$, where $v_i = lbF(V_i), \bar{v}_i = ubF(V_i)$ and $i = 1,2$.

Step 3 Discard the second triplet since $v_2 > 0$. Enter the first triplet into the list L at the end. Then remove the triplet $(Y \times [-1,1], y, \bar{y})$ from the list L.

Step 4 If some termination criterion holds, go to Step 7.

Step 5 Denote the first triplet of the list L by $(Y \times [-1,1], y, \bar{y})$.

Step 6 Go to Step 2.

Step 7 The algorithm ends.

It can be verified that, for $n \ge 1$,

$$L_n = \left\{\left(\left[-1, \tfrac{1}{2^n}\right] \times [-1,1], -1, \tfrac{1}{2^n}\right)\right\}, M_n = \left[-1, \tfrac{1}{2^n}\right] \times [-1,1],$$
$$M_n - X^* = \left(0, \tfrac{1}{2^n}\right] \times [-1,1], U_n = M_n = \left[-1, \tfrac{1}{2^n}\right] \times [-1,1].$$

Thus,

$$\lim_{n\to\infty} d(M_n) = \sqrt{5}, \lim_{n\to\infty} d(M_n - X^*) = 2, \quad \lim_{n\to\infty} \mu(M_n) = \lim_{n\to\infty} 2\left(1 + \frac{1}{2^n}\right) = 2,$$
$$\lim_{n\to\infty} \mu(M_n - X^*) = \lim_{n\to\infty} 2\left(\frac{1}{2^n}\right) = 0, \quad \lim_{n\to\infty} \mu_1(M_n - X^*) = 0,$$
$$\lim_{n\to\infty} \mu_2(M_n - X^*) = 2,$$

but $\cap_{n=1}^{\infty} U_n = \cap_{n=1}^{\infty} \left(\left[-1, \tfrac{1}{2^n}\right] \times [-1,1]\right) = [-1,0] \times [-1,1] = X^*$.

Remark 5.2 Example 5.1 tells us that C1a, C1b, C1c, and C1e are not necessary for the convergence of this particular DDA.

Example 5.2 $\left(P_3^1\right)$ Solve $x^2+y^2=0$ over $X=\{(x,y)|(x,y)\}\in$ Triangle $OA_{00}B_{00}$ with vertices $O(0,0), A_{00}(1,0)$, and $B_{00}(0,1)$.

It is easy to get the solution set of Problem $(P_3^1), X^*=\{(0,0)\}$. For simplicity, we describe the sets generated by Algorithm $\left(A_3^1\right)$ below according to their geometric features. For example, we use Triangle $OA_{00}B_{00}$ (or simply $OA_{00}B_{00}$) to represent the set X.

Algorithm (A_3^1)

Step 1 Set $Y=X=OA_{00}B_{00}$ and initialize the list $L=\{Y\}$.
Step 2 Split Y into two parts V_1 (on the left, a right triangle) and V_2 (on the right, a general triangle) along the line passing through B_{00} and the midpoint of the base line of Y on the x-axis. Discard V_2 and place V_1 at the end of the list L.
Step 3 If the sum of measures (i.e., areas) of all the sets in L is less than a small positive ε, then go to Step 6.
Step 4 Denote the first set of L by Y.
Step 5 Go to Step 2.
Step 6 The algorithm ends.

Based on Algorithm $\left(A_3^1\right)$, there are right triangles with one vertex at $O(0,0)$ as conditional sets in the lists. Now, we give lists generated by Algorithm (A_3^1) as follows, skipping the name of geometry of each set for simplicity.

$$L_n=\{OA_{n0}B_{00}\} \quad \text{with} \quad O(0,0), \quad A_{n0}\left(\frac{1}{2^n},0\right), \quad \text{and} \quad B_{00}(0,1) \quad \text{for } n\geq 1.$$

Thus, $U_n=OA_{n0}B_{00}$ for all $n\geq 1$. Since

$$OA_{n0}B_{00}\subseteq OA_{n-1,0}B_{00} \text{ for all } n\geq 2, \text{ and } OA_{n0}B_{00}-X^*\subseteq OA_{n0}B_{00} \quad \text{for all } n\geq 1,$$

$$\mu(OA_{n0}B_{00}-X^*)\leq \mu(OA_{n0}B_{00})=\frac{1}{2^{n+1}}\to 0 \text{ as } n\to\infty.$$

Hence, C1f is satisfied, and so are C2a, C3a, and C4a. On the other hand, since

$$\begin{aligned} d(OA_{n0}B_{00},OB_{00}) &= \max\{d_0(OA_{n0},B_{00},OB_{00}),d_0(OB_{00},OA_{n0}B_{00})\}\\ &= \max\{d_0(OA_{n0},B_{00},OB_{00}),0\}\\ &= d_0(OA_{n0}B_{00},OB_{00})\\ &= \frac{1/2^n}{\sqrt{1+(1/2^n)^2}}\to 0 \text{ as } n\to\infty, \end{aligned}$$

the sequence $\{OA_{n0}B_{00}\}_{n=1}^{\infty}$ converges to the segment OB_{00}. Thus, we get

$$\lim_{n\to\infty} U_n=\lim_{n\to\infty} OA_{n0}B_{00}=OB_{00}=\{(x,y)|x=0,y\in[0,1]\}\supseteq X^*=\{(0,0)\}.$$

Therefore, Algorithm $\left(A_3^1\right)$ does not converge.

Remark 5.3 Example 5.2 tells us that each of the following two sets of conditions is not sufficient for convergence of Algorithm $\left(A_3^1\right)$.

(a) C1e, C2a, C3a, and C4a.
(b) C1f, C2a, C3a, and C4a.

6 An Application of DDA

In this section, we apply the convergence analysis of our proposed DDA to one of the well-known interval algorithms, the Hansen's algorithm (cf. Hansen 1980; Ratscheck and Rokne 1988) for solving the standard unconstrained global optimization problem (P_1). Of course, we can also apply DDA to algorithms for solving other global problems.

In order to discuss the Hansen's algorithm, we introduce some special notations as follows.

- $X = [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_d, b_d] \subseteq \mathbf{R}^d$ is called an interval (box) of $\mathbf{R}^d$, where $a_j, b_j \in \mathbf{R}, j = 1, \ldots, d. w(X) = \max\{b_j - a_j : j = 1, \ldots, d\}$ is called the width of X, and $\text{mid}(X) = \text{mid } X = ((a_1 + b_1)/2, \ldots, (a_d + b_d)/2)$ the middle point of X.
- $\Diamond f(X) =$ the range of function $f: X \to \mathbf{R}$ over X.
- $w(\Diamond f(X)) =$ the width of the interval hull of $\Diamond f(X)$, which is the smallest compact interval that contains $\Diamond f(X)$.
- $I =$ the set of real compact intervals $[a, b] a, b \in \mathbf{R}$.
- $I(X) =$ the set of all interval boxes contained in X.
- $F: I(X) \to I$, an inclusion function of f over X. It is assumed to satisfy $w(F(Y)) \to 0$ as $w(Y) \to 0$.

6.1 Hansen's algorithm for solving global unconstrained minimization problem

The Hansen's algorithm was designed to solve (P_1). Here is one particular description of the algorithm.
Hansen's algorithm:

Step 1 Set $Y = X$ and initialize the list $L = \{(Y, y)\}$ with $y = lbF(Y)$. Set $\tilde{f} = ubF(c)$ with $c = \text{mid}(Y)$.
Step 2 Bisect Y along the direction perpendicular to an edge of the maximum length to get two boxes V_1, V_2 such that $Y = V_1 \cup V_2$.
Step 3 Enter the pairs (V_1, v_1) with $v_1 = lbF(V_1)$ and (V_2, v_2) with $v_2 = lbF(V_2)$ at the end of the list L and remove (Y, y) from the list L.
Step 4 Choose a pair $(\tilde{Y}, \tilde{y})$ in the list L which satisfies $\tilde{y} \leq z$ for all pairs (Z, z) of the list L.
Step 5 Discard all pairs (Z, z) from the list L that satisfy $\tilde{f} < z$ (the midpoint test).
Step 6 If a termination criterion holds, go to Step 8.
Step 7 Denote the first pair of the list L by (Y, y). Set $c = \text{mid}(Y)$ and $\tilde{f} = \min(\tilde{f}, ubF(c))$. Then go to Step 2.
Step 8 The algorithm ends.

6.2 Properties of the hansen's algorithm

According to the subdivision process of the Hansen's algorithm, along with the stated deletion condition, C2a clearly holds. Below, we verify three other conditions C1a, C3a, and C4a for the Hansen's algorithm, from which its convergence follows according to Corollary 5.1(a).

Property 6.1

(a) *If V is any box in $\mathbf{R}^d$, then $w(V) \leq d(V) \leq w(V)\sqrt{d}$.*
(b) *If $\{M_n\}_{n=1}^{\infty}$ is any sequence of conditional boxes generated by the Hansen's algorithm, then* $\lim_{n\to\infty} d(M_n) = 0$ *is equivalent to* $\lim_{n\to\infty} w(M_n) = 0$.

Proposition 6.1 *If $\{M_n\}_{n=1}^{\infty}$ is any sequence of conditional sets generated by the Hansen's algorithm, then* $\lim_{n\to\infty} w(M_n) = 0$.

Proof By the procedure of the Hansen's algorithm, it is clear to see

$$\lim_{n\to\infty} w(M_n) = 0.$$

□

Corollary 6.1 *If $\{M_n\}_{n=1}^{\infty}$ is any sequence of conditional sets generated by the Hansen's algorithm, then* $\lim_{n\to\infty} d(M_n) = 0$ (Cla).

Remark 6.1 Based on Property 6.1, we are able to measure boxes generated by the Hansen's algorithm by their widths instead of their diameters. In particular, we can replace $\lim_{n\to\infty} d(M_n) = 0$ with $\lim_{n\to\infty} w(M_n) = 0$ in Cla.

Remark 6.2 Since C1a is satisfied for the Hansen's algorithm, the other versions of Condition 1 are also satisfied according to Proposition 4.2.

For the remainder of this section, let

$Y_n =$ the leading box in $L_n, y_n = lbF(Y_n)$,
$\tilde{Y}_n =$ the box in L_n such that $lbF(\tilde{Y}_n) = \min_{(Z,z)\in L_n}\{lbF(Z)\}, \tilde{y}_n = lbF(\tilde{Y}_n)$,
$\hat{Y}_n =$ the box in L_n such that $ubF(\text{mid}(\hat{Y}_n)) = \min_{(Z,z)\in L_n}\{ubF(\text{mid}(Z))\}, \hat{y}_n = lbF(\hat{Y}_n)$.

Proposition 6.2 *Suppose V' is a conditional box of some list V'_n generated by the Hansen's algorithm. If $V' \cap X^* = \phi$, then, after a finite number of steps, V' and its sub-boxes will be discarded* (C3a).

Proof Suppose not. Then there is a conditional box $V'_n \subseteq V'$ in the list L_n for all $n > n'$ such that $V'_n \cap X^* = \phi$ and $V'_{n+1} \subseteq V'_n = \phi$. We also have $\lim_{n\to\infty} w\left(V'_n\right) = 0$ and $\lim_{n\to\infty} w(F(V'_n)) = 0$. Note that $V' \cap X^* = \phi$ implies $f^* \notin \Diamond f\left(V'\right)$. Thus, $f^* < lb\Diamond f(V')$ and $f^* < lb\Diamond f(V'_n)$ for all $n > n'$ since $\Diamond f(v'_{n+1}) \subseteq \Diamond f(V'_n) \subseteq f(V')$.

(1) Show $lbF(V'_n) \leq ubF(\text{mid}\tilde{Y}_n)$ for all $n > n'$ as follows.
(1a) If $V'_n = \tilde{Y}_n$, then, by $\text{mid}\tilde{Y}_n \in V'_n, F(\text{mid}\tilde{Y}_n) \subseteq F(V'_n)$ and $lbF(V'_n) \leq ubF$ $(\text{mid}\tilde{Y}_n)$.
(1b) If $V'_n \neq \tilde{Y}_n$, then $lbF(\tilde{Y}_n) \leq lbF(V'_n)$, no matter whether $\tilde{Y}_n$ is older or newer than V'_n in L_n.
For the case when $\tilde{Y}_n$ is older than V'_n in L_n, i.e., $\tilde{Y}_n$ is ahead of V'_n in L_n, we write

$$L_n = \left\{\ldots, \left(\tilde{Y}_n, \tilde{y}_n\right), \ldots, \left(V'_n, v'_n\right), \ldots\right\}$$

with $\tilde{y}_n = lbF(\tilde{Y}_n) \leq v'_n = lbF\left(V'_n\right)$. Note that, under our assumptions, V'_n is always in the subsequent lists until V'_n is bisected. From the above list, we have

$$L_{n_1} = \left\{\left(\tilde{Y}_n, \tilde{y}_n\right), \ldots, \left(V'_n, v'_n\right), \ldots\right\}$$

for some $n_1 > n$. Since $\tilde{Y}_n$ is the leading box of L_{n_1}, by the deletion condition, we have

$$v'_n \leq ubF(\mathrm{mid}\tilde{Y}_n), \text{ i.e., } lbF(V'_n) \leq ubF(\mathrm{mid}\tilde{Y}_n).$$

For the case when $\tilde{Y}_n$ is newer than V'_n in L_n, i.e., $\tilde{Y}_n$ is behind V'_n in L_n,

$$L_n = \{\ldots, \left(V'_n, v'_n\right), \ldots, \left(\tilde{Y}_n, \tilde{y}_n\right), \ldots\}.$$

Note that V'_n is kept in the following lists until V'_n is bisected. Together with $\tilde{y}_n \leq v'_n$, we have

$$L_{n_2} = \{\left(V'_n, v'_n\right), \ldots, \left(\tilde{Y}_n, \tilde{y}_n\right), \ldots\},$$

for some $n_2 > n$. We see that V'_n is the leading box of L_{n_2} and V'_n will be bisected into two sub-boxes V'_1 and V'_2. Under our assumption, at least one of them should be in the next list L_{n_2+1} and also in the subsequent lists until it is bisected. Let us say it is V'_1, to be specific. Thus,

$$L_{n_2+1} = \left\{\ldots, (\tilde{Y}_n, \tilde{y}_n), \ldots, \left(V'_1, v'_1\right)\right\}$$

with $v'_n = lbF\left(V'_1\right)$. According to the previous case, we get $lbF(V'_1) \leq ubF$ $(\mathrm{mid}\tilde{Y}_n)$. Note that $V'_1 \subseteq V'_n$ and $lbF(V'_n) \leq lbF(V'_1)$. Thus, $lbF(V'_n) \leq ubF(\mathrm{mid}\tilde{Y}_n)$ follows.

(2) We would like to find out some results that would lead to a desired contradiction.

(2a) After a finite number of steps, $lb\Diamond f(V') > ubF(\tilde{Y}_n)$; otherwise, $lbF(\tilde{Y}_n) \leq f^* < lb\Diamond f(V') \leq ubF(\tilde{Y}_n)$. Thus, $\lim_{n\to\infty} w(F(\tilde{Y}_n)) \geq lb\Diamond f(V') - f^* > 0$, contradicting to $\lim_{n\to\infty} w(F(\tilde{Y}_n)) = 0$.

(2b) $\lim_{n\to\infty} lbF(V'_n) = f^*$ since $f^* \in F(\tilde{Y}_n), \lim_{n\to\infty} w(F(V'_n)) = 0$, and V'_n is in L_n with $ubF(\tilde{Y}_n) \geq ubF(\mathrm{mid}\tilde{Y}_n) \geq lbF(V'_n) \geq lbF(\tilde{Y}_n)(n > n')$.

(2c) $\lim_{n\to\infty} lbF(V'_n) = lb\Diamond f(V')$, because, after a finite number of steps, we have $ubF(V'_n) \geq lb\Diamond f(V'_n) \geq lb\Diamond f(V') > ubF(\tilde{Y}_n)ubF(\mathrm{mid}\tilde{Y}_n) \geq lbF(V'_n)$.

(2d) Note that $f^* < lb\Diamond f(V')$, (2b) and (2c) do contradict to each other. Thus, the proof is complete. □

Proposition 6.3 *If the inclusion function F of f satisfies $w(F(Y)) \to 0$ as $w(Y) \to 0$, then the solution set $^*X$ is closed (C4a).*

Proof If X^* is finite, then X^* is clearly closed. In case that X^* is infinite, note that $w(F(Y)) \to 0$ as $w(Y) \to 0$ for $Y \in I(X)$ implies that $f(x)$ is continuous over X. Let $\left\{x^*_k\right\}_{k=1}^{\infty}$ be any convergent sequence in X^* with $\lim_{n\to\infty} x^*_k = x'$. Then

$$f(x^*_k) = f^* \quad \text{for all } k \geq 1 \text{ and } f(x') = \lim_{n\to\infty} f\left(x^*_k\right) = f^*.$$

Thus, we have $x' \in X^*$. Therefore, X^* is closed. □

6.3 Convergence of the hansen's algorithm

Based on the above verification of four convergence conditions of the Hansen's algorithm, it is clear to see that the Hansen's algorithm is convergent under the assumption that the inclusion function F of f satisfies $w(F(Y)) \to 0$ as $w(Y) \to 0$. Without applying DDA, a different proof of the convergence of the Hansen's algorithm was given in Ratscheck and Rokne (1988).

7 Discussion

In this paper, we have presented a general prototype of convergent algorithms (called DDA) for locating all the solutions of the most general global problem. The DDA has four essential ingredients, subdivision, deletion, selection, and stopping conditions. Although DDA is stated in a sequential fashion, it is obviously capable of taking full advantage of parallel implementation. Its convergence is proved to be characterized by four categories of conditions. For the convergence of DDA, we have verified four sets of sufficient conditions, nine necessary conditions, and four sets of necessary and sufficient conditions. Our results provide useful guidelines for developing new implementations of global search algorithms, and reliable theoretical justifications for their convergence as well as convergence of some existing algorithms. The paper uses a well-known Hansen's interval algorithm as an example of implementation with implied convergence. We are working on interval implementations for solving some specific kinds of global search problems mentioned in Sect. 2. In particular, we hope that this research would lead to a series of new results in the global optimization and global search in general.

Acknowledgements The authors are grateful to the referees for the helpful comments on an earlier version of this paper.

References

Alefeld, G., Herzberger, J.: Introduction to Interval Computations. Academic Press, New York, NY. (1983)
Csallner, A.E., Csendes, T., Markót, M.C.: Multisection in interval branch-and-bound methods for global optimization I. Theoretical results. J. Global Optim. **16**, 317–392 (2000)
Csendes, T.: New subinterval selection criteria for interval global optimization. J. Global Optim. **19**, 307–327 (2001)
Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA (1989)
Hansen, E.R.: Global optimization using interval analysis: the multidimensional case. Numer. Math. **34**, 247–270 (1980)
Hansen, E.R.: Global Optimization Using Interval Analysis. Marcel Dekker, NY (1992)
Horst, R.: An algorithm for nonconvex programming problems. Math. Program. **10**, 312–321 (1976)
Horst R., Tuy H.: Global Optimization, Deterministic Approaches. Springer-Verlag, Berlin (1990)
Kearfott, R.B.: A review of techniques in the verified solution of constrained global optimization problems. Applications of interval computations (El Paso, TX, 1995), Appl. Optimi. **3**, 23–59 (1996)
Kirkpatrick, S., Gelatt, Jr. C.D., Vecchi, M.P.: Optimization by simulated annealing, Science **220**, 671–680 (1983)
Lawler, E.L., Wood, D.E.: Branch-and-bound methods: A survey. Oper. Res. **14**, 699–719 (1966)
Moore, R.E.: Methods and Applications of Interval Analysis, SIAM Publication. Philadelphia, Pennsylvania (1979)

Neumaier, A.: Interval Methods for Systems of Equations. Cambridge University Press, Cambridge, UK (1990)
Ratscheck, H., Rokne, J.: New Computer Methods for Global Optimization. Wiley, New York, NY (1988)
Royden, H.L.: Real Analysis, Macmillan. New York, NY (1968)
Rudin, W.: Principles of Mathematical Analysis, 2nd edn. McGraw-Hill Inc., New York, NY (1964)
Sun, M.: Tree annealing for constrained optimization, Proceedings of 34th IEEE Southeastern Symposium on System Theory, pp. 412–416 Huntsville, AL, (2002)
Sun, M., Johnson, A.W.: Interval branch and bound with local sampling for constrained global optimization, J. Global Optimi. **33**, 61–82 (2005)
Xu, Z.B., Zhang, J.S., Leung, Y.W.: A general CDC formulation for specializing the cell exclusion algorithms of finding all zeros of vector functions. Appl Math Comput **86**, 235–259 (1997)