# A Review of Techniques in the Verified Solution of Constrained Global Optimization Problems

R. Baker Kearfott

Elements and techniques of state-of-the-art automatically verified constrained global optimization algorithms are reviewed. These include, among other things, a description of ways of rigorously verifying feasibility for equality constraints and a careful consideration of the role of active inequality constraints. Previously developed algorithms and general work on the subject are also listed. Limitations of present knowledge are mentioned, and advice is given on which techniques to use in various contexts.

The basic problem to be considered is

$$\begin{array}{lrcll} \text{minimize} & \phi(X) & & & \\ \text{subject to} & c_i(X) & = & 0, & i = 1,\ldots,m \\ & a_{i_j} \leq & x_{i_j} & \leq b_{i_j}, & j = 1,\ldots,q, \end{array} \tag{1}$$

where $X = (x_1, \ldots, x_n)^T$.

Specific interval global optimization algorithms are considered to be specific cases of a general pattern. The basic ideas in this pattern include a *check on the range* of $\phi$ and a *computational existence / uniqueness* test. We let $\mathbf{X} = (\mathbf{x}_1, \ldots \mathbf{x}_n)^T = ([\underline{x}_1, \overline{x}_1], \ldots, [\underline{x}_n, \overline{x}_n])^T$ be the interval vector ("box") representing the search region $\underline{x}_i \leq x_i \leq \overline{x}_i$, $1 \leq i \leq n$. Such algorithms can maintain a list $\mathcal{L}$ of boxes $\mathbf{X}$ to be processed, a list $\mathcal{U}$ of boxes the algorithm has reduced to small diameter, and a list $\mathcal{C}$ of boxes that have been verified to contain critical points. The pattern can then be stated as:

**Algorithm 1** (Abstract Branch-and-Bound Pattern)

1. *Initialize $\mathcal{L}$ by placing the initial search region $\mathbf{X}_0$ in it.*
2. DO WHILE $\mathcal{L} \neq \emptyset$.
   (a) *Remove the first box $\mathbf{X}$ from $\mathcal{L}$;*
   (b) (Process $\mathbf{X}$) *Do one of the following:*
      - *reject $\mathbf{X}$;*
      - *reduce the size of $\mathbf{X}$;*
      - *determine that $\mathbf{X}$ contains a unique critical point, then find the critical point to high accuracy.*
   (c) *Insert one or more boxes derived from $\mathbf{X}$ onto $\mathcal{L}$, $\mathcal{U}$ or $\mathcal{C}$, depending on the size of the resulting box(es) from step 2b and whether the (possible) computational existence test in that step has determined a unique critical point.*

**End Algorithm 1**

(Possibly optional) acceleration devices used in Algorithm 1 can be listed as:

**Algorithm 2** (Range Check and Critical Point Verification)

1. *Input a box $\mathbf{X}$ and the current best rigorous upper bound $\overline{\phi}$ on the global minimum.*
2. (Feasibility check; for constrained problems only)
   (a) (Exit if infeasibility is proven) DO *for $i = 1$ to $m$:*
      i. *Compute an enclosure $c_i(\mathbf{X})$ for the range of $c_i$ over $\mathbf{X}$.*
      ii. IF $0 \notin c_i(\mathbf{X})$ THEN EXIT.
   (b) *Prove computationally, if possible, that there exists at least one feasible point in $\mathbf{X}$.*
3. (Range check or "midpoint test")
   (a) *Compute a lower bound $\underline{\phi}(\mathbf{X})$ on the range of $\phi$ over $\mathbf{X}$.*
   (b) IF $\underline{\phi}(\mathbf{X}) < \overline{\phi}$ THEN EXIT.
4. (Update the upper bound on the minimum.) IF *the problem is unconstrained or feasibility was proven in step 2b,* THEN
   (a) *Use interval arithmetic to compute an upper bound $\overline{\phi}(\mathbf{X})$ of the objective function $\phi$ over $\mathbf{X}$.*
   (b) $\overline{\phi} \leftarrow \min\{\overline{\phi}, \overline{\phi}(\mathbf{X})\}$.
5. ("monotonicity test")
   (a) *Compute an enclosure $\nabla\phi(\mathbf{X})$ of the range of $\nabla\phi$ over $\mathbf{X}$. (Note: If $\mathbf{X}$ is "thin,", i.e. if some bound constraints are active over $\mathbf{X}$, then a* reduced gradient *can be used.*

Department of Mathematics, University of Southwestern Louisiana, U.S.L. Box 4-1010, Lafayette, Louisiana 70504-1010.
This work was supported in part by National Science Foundation grant CCR-9203730

*(b)* IF $0 \notin \nabla\phi(\mathbf{X})$ THEN EXIT.

6. ("concavity test") *If the Hessian matrix*[1] $\nabla^2\phi$ *cannot be positive definite anywhere in* $\mathbf{X}$ THEN EXIT.
7. (Quadratic convergence and computational existence / uniqueness) *Use an interval Newton method*[2] *(with the Fritz–John conditions in the constrained case) to possibly do one or more of the following:*
   - *reduce the size of* $\mathbf{X}$*;*
   - *discard* $\mathbf{X}$*;*
   - *verify that a unique critical point exists in* $\mathbf{X}$.
8. (Bisection or geometric tesselation) *If step 7 did not result in a sufficient change in* $\mathbf{X}$*, then bisect* $\mathbf{X}$ *along a coordinate direction (or otherwise tesselate* $\mathbf{X}$*), returning all resulting boxes for subsequent processing.*

**End Algorithm 2**

Within the review paper, nine published algorithms are mentioned in tables, giving information on which of the devices in Algorithm 2 are used, as well as information concerning the extent of reported numerical experiments.

Elements of such algorithms are also reviewed. In particular, choices involved in constructing interval Newton methods are outlined. These include choice of the method itself, choice of method of computing the derivative matrix, choice of preconditioner, choice of base point, and the form of the Fritz–John conditions for problem 1. A unique part of this discussion is an observation concerning how to combine a technique of Hansen with computations of interval slopes; this discussion appears along with simple examples. Also, careful consideration is given of properties and choices for different goals, such as:

- reducing the size of regions $\mathbf{X}$, with quadratic convergence of the widths to zero;
- computationally (but rigorously) proving existence of solutions within $\mathbf{X}$;
- proving that that there is a unique solution within $\mathbf{X}$;
- proving that there can be no solutions in $\mathbf{X}$.

With regard to the treatment of constraints, differences in between techniques that

- prove approximate feasibility,
- rigorously prove feasibility, or
- rigorously prove infeasibility

are clarified. In particular, the need to rigorously prove feasibility is explained, and a technique for rigorously proving feasibility of equality constraints, related to a technique of Hansen and explained in a concurrent paper of the author, is reviewed.

There is also an extensive section on handling bound constraints. Several diagrams illustrate a tesselation process for bound constraints, first appearing in a previous paper of the author, while an algorithm for it is given in simple recursive form. The process can be viewed as traversing a ternary tree. Implications concerning computational complexity of bound constraints are discussed, and techniques for rigorously pruning the tree are mentioned. The techniques are advocated, along with use of slack variables, for general inequality constraints, as well as bound constraints.

Finally, there is a section detailing applications to date

Sixty-six references appear.

**Key words.** constrained global optimization, verified computations, interval computations, bound constraints.

**AMS subject classifications.** 65K05, 90C26

[1] or reduced Hessian matrix, as in step 5

[2] possibly in a subspace, as in steps 5 and 6