Reliable Computing, 0, 1–16 (0)

Rigorous Global Search: Industrial Applications *

GEORGE F. CORLISS georgec@mscs.mu.edu
Department of Mathematics, Statistics, and Computer Science, Marquette University, P.O. Box 1881, Milwaukee, WI 53201–1881

R. BAKER KEARFOTT rbk@usl.edu
Department of Mathematics, University of Southwestern Louisiana, U.S.L. Box 4–1010, Lafayette, LA 70504–1010

Received October 30, 1998; Revised Somewhat later

Editor: T. Csendes

Abstract. We apply interval techniques for global optimization to several industrial applications including Swiss Bank (currency trading), BancOne (portfolio management), MacNeal-Schwendler (finite element), GE Medical Systems (Magnetic resonance imaging), Genome Theraputics (gene prediction), inexact greatest common divisor computations from computer algebra, and signal processing. We describe each of the applications, discuss the solutions computed by Kearfott's GlobSol software (see `www.mscs.mu.edu/~globsol`), and tell of the lessons learned. In each of these problems, GlobSol's rigorous global optimization provided significant new insights to us and to our industrial partners.

Keywords: global optimization, currency trading, portfolio management, least common denominator, interval computations.

1. Problem domain

Let f be a continuous, differentiable objective function $X \subset \mathbf{R}^n \to \mathbf{R}$. Consider the global optimization problem,

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

possibly with linear or nonlinear constraints. We seek validated, tight bounds for the set of minimizers X^* and/or the optimum value f^*. Linear systems $Ax = b$ and nonlinear systems $G(x) = 0$ for $G : \mathbf{R}^n \to \mathbf{R}^n$, are also solved by components in our tool box. The rigorous global optimization algorithm[10, 15] uses an exhaustive branch and bound search. It discards regions where the solution is guaranteed *not* to be, it attempts to validate the existence of a unique local minimum in a box, and it makes the enclosing box tight. In contrast, conventional approximate algorithms for global optimization include exhaustive search, simulated annealing [11], and genetic algorithms [1].

The validated global optimization algorithm used by the GlobSol software package is related to algorithms described by Hansen [10], Neumaier [18], Ratz [20, 21, 22],

* This work is supported in part by Sun Microsystems and by the National Science Foundation under various grants.

Jansson [13, 14], or van Iwaarden [12]. Details of the GlobSol algorithm appear in [15], an overview appears in [5], and details of the applications described here are available from the GlobSol web site: `www.mscs.mu.edu/~globsol`

Features contributing to the difficulty of a general global optimization problem include

Feasibility: Is there a feasible solution? Interval techniques allow us safely to discard regions where the solution cannot be and to enclose regions guaranteed to be feasible.

Many local critical points: How can we tell whether the local minimum we have found really is the global minimum? For an objective function with many local minima, like an egg carton, interval arithmetic can guarantee that a local minimum is indeed a true global minimum.

Singular or near-singular solutions: A nearly flat objective function, like the top of an egg carton, exhibits singular (or nearly singular) behavior. Interval techniques can bound the entire set of points X^* at which the global minimum is attained.

Uncertain parameters: Many models contain parameters whose value is not precisely known. Interval techniques automatically handle interval-valued parameters.

High dimensions: The algorithm is an exhaustive branch and bound search to guarantee that the global optimum cannot be missed. The code tries to be clever, but the worst-case complexity is $O(2^n)$ for a problem with n independent variables. The complexity is also exponential in the requested tolerance. For some problem classes, the observed complexity is $O(n^3)$ governed by the linear algebra. The largest problem we have successfully handled so far is $n = 128$ (a mildly nonlinear trayed tower model solved by Schnepper, a student of Stadtherr, taking special account of sparsity). Work is in progress with parallel architectures, better heuristics, and sparsity to handle larger problems.

The point of this paper is to outline briefly several actual industrial applications that have been addressed using the GlobSol software. The intent is to convey the wide domain of applicability of the rigorous global search methods. More details about each application are available from technical reports cited here and may appear in subsequent papers.

2. Swiss Bank (Currency Trading)

Swiss Bank's portfolio is managed using a proprietary risk control system. This system aggregates the currency trades and determines an overall value of the portfolio. The total value of the portfolio is evaluated in the light of several risk factors. The objective of the system is to determine the value at risk, or the maximum overnight loss the bank may face on the portfolio of trades, assuming that the portfolio is illiquid overnight.

The maximum loss is determined for the portfolio closing position by allowing the risk factors to change and calculating the price sensitivities of the trades to these changes in the risk factors. As the risk factors are allowed to change within specified ranges, the calculation of the maximum loss becomes quite difficult. The theoretical shape of the portfolio frontier becomes more complex as the number of currencies and risk factors increases. Therefore, it becomes important to guarantee that a global maximum is found, as a local maximum may leave a great deal of risk hidden.

Value at Risk (VaR) is a measure of the maximum loss in market value over a given time interval within a given confidence interval. That is, the loss of market value that is exceeded by a given probability. The Bank for International Settlements, in determining adequate bank capital, suggests a 99 percent confidence interval and a ten-day time interval. Given a 99% confidence interval, the VaR approach yields a single estimate of potential loss that would be exceeded in value with only 1 percent probability. As the VaR methodology has gained international acceptance, it has become more important to banks as a risk management tool. Swiss Bank plans to expand its use of the methodology and improve their current approaches. By being able to accurately assess the maximum expected overnight loss exposure, the Bank will be in a much better position to adjust its holdings to reduce that potential loss and satisfy regulatory requirements.

GlobSol was tested using the Garman-Kohlhagen model for valuing foreign exchange options and for determining the VaR for portfolios containing calls and puts on 2, 4, and 7 exchange rates. Initial experiments with the options model suggest that the CPU time for these problems grows exponentially with the number of instruments in the portfolio. In these models, market forces work to make the objective functions nearly flat, with many small relative local optima. The guarantee of GlobSol's solutions is that we are certain to have found the best of the local optima.

For the Value at Risk model, we solve the problem of finding the VaR of a portfolio. We claim that not only the maximum loss of the given portfolio containing any finite number of instruments is found, but also guarantee that it is the global maximum that is enclosed in tight bounds.

We used GlobSol to find the minimum variance portfolio for one two week period in October, 1997, of a portfolio containing Swiss Franks, Mexican Pesos, South Korean WONs, Australian Dollars, Netherlands' Guilders, and Singapore Dollars. The minimum variance portfolio had 74.7% Mexican Peso, 8.9% Australian Dollar, 8.4% Netherlands' Guilder, and 8.0% Singapore Dollar. The minimum variance for the two week period was 0.00055% of the portfolio value, while the variances of the individual currencies ranged from 0.0013% to 0.049%.

The Swiss Bank application is joint work with Shirish Ranjit, Joe Daniels, and Peter Toumanoff. Details appear in [6, 19] and at
`www.mscs.mu.edu/~globsol/Marquette/apSwiss.html`

3. BancOne (Portfolio Management)

We consider the universal portfolio management problem: Given various investment constraints (cash flow, risk element, maturity structure, corporate investment policies, etc.), what combination of securities will maximize return? This concept can be most easily related to efficient frontier and security market line solutions of classic finance. Although quite simple in theory, a solution is difficult to implement for several reasons:

- Managers may be responsible for several portfolios, each with different investment policies and objectives. This renders the efficient frontier of investments subordinate to investment policy.
- Selecting securities from the universe of acceptable investments, pricing options, hedging strategies, even for the most basic fixed income portfolio, can require substantial amounts of time.
- Statistics used to judge risk and return are often time consuming to calculate, numerous, and of uncertain accuracy. Therefore, these statistics are often not considered completely.

For these and other reasons, rule-of-thumb investment management often substitutes for more theoretically sound investment portfolio management and asset techniques. For instance, securities may be selected because they fall within the investment policy of the portfolio, or because the security matches the historical return or average life of the portfolio.

The required software solution to the problem first involves selecting securities from a large universe of fixed income securities which meet the required investment parameters (i.e. the investment policy of the fund). Then, given these securities, select the optimal mix (or asset allocation) to maximize the expected return of the portfolio while meeting bank policy objectives regarding the duration to maturity, cash flow, and risk. In addition, given the selected optimal portfolio(s), the software solution demonstrates to the money manager the additional opportunities and risks which exist by relaxing one or several of the policy constraints.

The solution requires flexibility to accommodate a wide range of investment styles. For instance, risk may be defined in several ways: credit risk, interest rate risk, variance in return, maturity, etc. Similarly, return may be defined as cash flow yield, coupon yield, total return, capital appreciation, etc. Finally, the final solution also addresses some of the usual portfolio constraints, including the following: tolerance for duration and convexity; required cash flow timing; selected average life of the portfolio; coupon yields or maturity dates; credit ratings; selected responsiveness to changes in partial duration.

A sample fixed-income portfolio is divided into two different models for this analysis. First, we model a portfolio containing only 3-, 6-, and 12-month Treasury bills. To simulate a real-world Treasury bill portfolio, we used actual spot price data for each type of Treasury bill from a recent U.S. Treasury auction. Historical spot price data for each bill was used to calculate the variances of each type of T-bill and covariances between bills. Using a typical set of constraints, GlobSol calculated the optimal allocation to be about 10% in 3-month T-bills, 60% in 6-month T-bills, and 30% in 12-month T-bills, for an average rate of return of 5.363% per year.

Second, we model a portfolio containing only 1-, 5-, 10-, and 30-year Treasury bonds. The objective in each case is to maximize the portfolio's rate of return, given four investment constraints of average duration, cash flow, variance, and total value. For the Treasury bond portfolio model, GlobSol computes the maximum portfolio rate of return subject to four investment constraints. Treasury bond variances and covariances between bonds were calculated using historical data. Hypothetical values were chosen for spot prices, coupon payments, and face values. We computed the rate of return for each bond. For a typical set of constraints, GlobSol determined the optimal allocation to be about 4% in 1-year bonds, 65% in 5-year bonds, 4% in 10-year bonds, and 26% in 30-year bonds. This allocation generated an optimal rate of return of 7.388%.

The results generated by GlobSol guarantee that 1) All constraints are satisfied, and 2) There is no allocation satisfying the investment constraints that offers a higher yield. Guaranteeing that we have determined the optimal allocation is a significant statement to make. This implies that we have assured that all constraints are satisfied. From the perspective of the portfolio manager, this is a guarantee that all portfolio objectives are being met. With financial market volatility constantly changing the value, risk, and return of the portfolio, actual portfolio objectives may not be being met even if the manager is active in trying to meet them. Therefore, the manager may not be able to make this assertion on a consistent basis. The results from GlobSol imply that we can say with 100% certainty that the constraints are being satisfied. Most importantly, the generated allocation of securities contains the optimal rate of return given the constraints. The manager can be assured that potential returns are not being missed, and shareholders are benefitting from optimal portfolio management.

The Banc One application is joint work with Paul Thalacker, Joe Daniels, Peter Toumanoff, Kristie Julien, and Ken Myszka. Details appear in [23] and at `www.mscs.mu.edu/~globsol/Marquette/apBancOne.html`

4. MacNeal-Schwendler (Finite Element Analysis)

A rocket engine exhaust nozzle's primary function is to direct the outward flow of high-pressure exhaust gas produced by the engine. In this manner, the thrust of the engine is focused, thus achieving the jet propulsion necessary to drive the vehicle forward.

While in operation, a rocket engine nozzle cone experiences an enormous internal material stress due to the massive thrust pressure and temperature created by the ignition of the rocket fuel within the engine. This material stress tends to act on the nozzle cone in a circumferential direction, trying to blow the nozzle apart. This type of material stress is exactly the same stress that causes a toy balloon to rupture from excessive internal air pressure. The task of the rocket nozzle designer is to design a nozzle cone that is strong enough to withstand the extreme forces and temperatures generated within the nozzle during engine operation, while obeying all design specifications.

Design problems like the rocket engine nozzle stress pose incredible challenges to aerospace and materials engineers. To reliably solve these difficult problems in very complex shapes, engineers employ the help of several analysis tools to accurately model the behavior of their designs on a computer before a prototype is actually constructed. One of the most powerful numerical analysis tools utilized by designers is the Finite Element Method.

We decided to pursue the first-principles finite element analysis, or the "all-together" approach, in which the finite element equations are presented as constraints to the optimization problem. Only values satisfying the FEA equations are considered feasible designs over which optimization can be performed. The method takes its name from a technique proposed by John Dennis and Karen Williamson for differential equation parameter identification problems. To gain experience with the all-together finite element approach, we first attacked a VERY simple structural analysis problem, minimizing the maximum stress an axially loaded elastic bar.

We have successfully modeled a 97 node axially loaded bar with the all-together approach. The technique begins with a model consisting of four nodes and three elements. GlobSol solves this problem and returns a solution consisting of four tightly bound node intervals. The results of this model run are then used to build a new model consisting of twice the number of elements. GlobSol now gets half of the model nodes bounded tightly, and once again GlobSol solves the problem. This procedure is repeated until we decide to stop at a model consisting of 96 elements and 97 nodes. The total CPU time required by GlobSol to complete the entire series was about 2.5 hours. This model run is significant because it demonstrates that GlobSol can solve a problem based on the all-together technique. Our primary concern is CPU time required.

The MacNeal-Schwendler application is joint work with Frank Fritz, Andy Johnson, Don Prohaska, and Bruce Wade. Details appear in [9, 8] and at `www.mscs.mu.edu/~globsol/Marquette/apMacNeal.html`

5. GE Medical Systems (Magnetic Resonance Imaging)

MRI Imaging Instrument is one of the major sophisticated, state-of-the-art, and expensive medical imaging products manufactured by GE Medical Systems. Through a clinical experiment, 3D medical images of a patient are generated by the principle of Nuclear Magnetic Resonance (NMR). One of the key components in the MRI is the MRI coil, which generates the electro-magnetic fields around the patient. The coil is stimulated by current impulses of a certain shape, which are expected to generate an electro-magnetic field of similar shape. However, due to principles of physics, the current impulse is not uniformly distributed in the coil. The current passing through the edge of the coil is greatly distorted. This is called the "Eddy Current" effect, which causes the distortion of the corresponding electro-magnetic fields, resulting a distorted medical image. Thus this eddy current effect must be corrected or compensated using the pre-generated compensation current.

The problem is formulated as a sequence of parameter identification problems:

Linear Model

We have a set of inputs (experimental $f(u,t)$) and their associated outputs (experimental $G(t)$). Initially, we assume the model F to be described by a linear, constant coefficient differential equation

$$x'' + a_1 x' + a_2 x = a_3 f(u,t).$$

Then the optimization problem is
Independents: a_1, a_2, a_3
Objective: Sum over all samples $|G(t) - x(t)|_2$

This is an example of a general parameter identification problem. We also consider fitting in a uniform or weighted sense.

General Model

We do not expect the linear model to fit sufficiently well, so we will explore nonlinear parameterized differential equations

$$x' = F(x, t, f(u,t), a).$$

Then the optimization problem is
Independents: a
Objective: Sum over all samples $|G(t) - x(t)|_2$.

Discover the Driving Function

Once we have an appropriate model F, we will proceed to determine an optimal driving function $f(u,t)$. For example, we may assume f can be expressed as sum of exponential functions:

$$f(u,t) = \sum \alpha_i \exp(-t/\tau_i),$$

and use optimization to find the best parameters. The optimization problem is
Independents: a_i, τ_i
Objective: $|G(t) - x(t)|_2$.

In each case of interest, we wish to determine parameters such that the solution to a differential equation fits a target, so we consider the general dynamical systems parameter identification problem. Given a set of data (t_i, x_i), or a target function $G(t)$, and a differential equation

$$x' = f(t, x, a),$$

determine values for the parameters a such that $x(t)$ best fits:

$$\min |G(t) - x(t)|_2.$$

Independents:

a	parameters in the DE,
c_j	coefficients of the basis functions b_j, and
x_i	approximate solution at the nodes.

Objective: Sum over all samples $|G(t) - x(t)|_2$
Constraints: $\sum c_j b_j'(t_i) = f(t, \sum c_j b_j(t_i), a)$.

We have solved one example of the second order linear ODE parameter identification problem with 20 nodes using the interval all-together approach, demonstrating the promise of the approach. However, we have not yet been able to scale up our initial success because the differential equation is moderately stiff, the parameters have widely different scales, and the system of constraint equations is poorly conditioned.

The GE Medical Systems application is joint work with Xin Feng, Ruoli Yang, Yunchuan Zhu, Yonghe Yan, and Robert Corless. Details appear in [7] and at `www.mscs.mu.edu/~globsol/Marquette/apGEMed.html`

6. Genome Theraputics (Gene Prediction)

All mammalian genes discovered to date have revealed a pattern to how their information is transcribed and translated. This pattern of information or structure of a gene can form a basis of an intelligent search of a similar DNA sequence or collection of sequences. This premise forms the basis for the identification of a gene by a process of pattern recognition using multiple genomic features from the different regions of a gene (Promoter and Structural regions). Over the last fifteen years, many molecular biologists, mathematicians, and computer scientists have attempted to develop numerical and non-numerical programming methods to analyze, understand, and decode the genetic information within DNA. As methods in molecular genetics expanded, the sequencing of DNA from a variety of species grew dramatically, the structure and function of genes were discovered, and programs and methods have evolved to discover the pattern of information in a gene.

Consider a researcher with a DNA sequence of interest. The sequence is submitted to several gene search engines available on the Internet, and each returns an indication of the genes it recognized and perhaps an indication of its confidence. Usually, the results from different gene search engines are not in perfect agreement. The researcher must reconcile the conflicting answers before beginning expensive laboratory analysis.

Our goal is a meta-engine to

- Accept the researcher's DNA sequence,
- Submit it to multiple Internet gene search engines,
- Gather the results,
- Reconcile the conflicting results, and

- Report a "consensus" answer.

The Meta Gene Prediction Engine will use a neural network trained by the GlobSol global optimization software. The goal is to extract more accurate gene information on stretches of DNA from existing gene prediction systems.

The objective function for training a general neural network, whether we use an L_2 or an L_∞ norm, is nearly flat or even singular. Since we do not know in advance the architecture of a net with power sufficient for the problem at hand, we often try a large network. The resulting optimization problem may be over-parameterized. Conventional techniques search down-hill until they find a local minimum, but starting at different points often yields wildly different parameter values for the neural network.

Validated optimization techniques can reliably distinguish local from global minima and can characterize sets of equally good parameter values. However, characterization of sets of solutions is very CPU intensive, so we have only handled very small neural networks. Work is continuing to improve the performance of GlobSol for sets of solutions. A prototype of the Meta Gene engine is on the web at `ares.ifrc.mcw.edu/MetaGene`. This prototype does not use GlobSol, but it does use optimization techniques.

The Genome Theraputics application is joint work with Zhitao Wang and Peter Tonellato. Details appear in [24] and at
`www.mscs.mu.edu/~globsol/Marquette/apGeneome.html`

7. Inexact Greatest Common Divisor

Given two polynomials $p(x)$ and $q(x)$ with inexact floating-point coefficients, how can we compute their greatest common denominator $d(x)$ (GCD)? The GCD is the polynomial of highest degree that divides both $p(x)$ and $q(x)$ exactly. The GCD is unique, up to multiplication by a constant.

If the polynomials are assumed to be known exactly, the GCD can be computed exactly by the Euclidian algorithm, or a variant, as is done by any current computer algebra (CA) system. However, if the coefficients are inexactly known (e.g. floating-point numbers), the polynomials are almost surely relatively prime, and their GCD is 1. This answer is true, but not helpful. Most CA users want a polynomial of as high degree as possible that is the GCD of a pair of 'nearby' polynomials [4].

If $p(x)$ and $q(x)$ are two (or more) univariate polynomials with inexact floating-point coefficients, then an $\epsilon \geq 0$ GCD of p and q is a polynomial $d(x)$ of highest possible degree such that $d(x)$ exactly divides $p(x) + \Delta p(x)$ and $q(x) + \Delta q(x)$ for some Δp and Δq with $||\Delta p(x)|| \leq \epsilon$ and $||\Delta q(x)|| \leq \epsilon$. We use the 2-norm of the vector of coefficients of a polynomial.

We have successfully applied GlobSol to several example problems from the literature, see [3]. For example, let

$$p(x) := x^5 + 5.503x^4 + 9.765x^3 + 7.647x^2 + 2.762x + 0.37725$$
$$q(x) := x^5 - 2.993x^4 - 0.7745x^3 + 2.007x^2 + 0.7605x$$

from [4]. GlobSol found

$$d(x) \in x^2 + [1.\underline{0069986}925, 1.\underline{0069986}978]x + [0.\underline{2534949}53, 0.\underline{2534949}61]$$

with $||\Delta p(x)||^2 + ||\Delta q(x)||^2 \approx 8 \times 10^{-7}$ in about 160 CPU seconds on a SPARC 10. For this example, giving GlobSol the correct answer as an initial guess made no difference in execution time.

The inexact greatest common divisor application is joint work with Paulina Chin and Robert M. Corless. Details appear in [2, 3] and at
`www.mscs.mu.edu/~globsol/Marquette/apGCD.html`

8. Signal Processing

Parameter estimation plays an important role in many areas of scientific and engineering computation including system identification for the design of control systems, pattern recognition systems, equalization of communications channels, and artificial neural networks. In each of these applications, parametric models are used to either emulate an unknown system, perform a specific task within the system, or mitigate anomalies caused by uncertainty. Conventional estimation methods are often used to compute the parameters so that the mean-squared error between the model output and the desired response is minimized.

Kelnhoffer [17] used interval global optimization techniques very similar to those used by GlobSol to construct bounded response models. That is, he computed intervals of parameter values that are consistent with model data. He applied his technique to problems from system identification, pattern recognition, channel equalization, and artificial neural networks.

One exciting aspect of Kelnhoffer's work is his interpretation of the results as model validation. If the interval optimization technique returns a solution (a set of parameter values consistent with the data), it has validated the model. Often more helpful, though disappointing, the technique often returns no solution. In that case, we have validated that the model is *inconsistent* with the data. The model builder must construct a new model. Interval techniques have this unique property of helping the modeler stop trying to work with the wrong model.

The details of Kelnhoffer's signal processing application appear in [17].

9. On GlobSol's Algorithm

Here, a brief review of the principles of interval branch and bound algorithms is given, as well as an overview of some unique aspects of the GlobSol package. Detailed descriptions appear in [15], [16], or from the World Wide Web page
`www.mscs.mu.edu/~globsol/User_Guide/13WhatIs/`. The GlobSol working note [5] includes an elementary introduction to the techniques in interval branch and bound algorithms for global optimization.

9.1. Branch and Bound

With interval arithmetic to compute rigorous range bounds and automatic differentiation to compute derivatives, the algorithm of GlobSol proceeds by branch and bound. We maintain three lists of boxes:

$\mathcal{L}$ not fully analyzed;
$\mathcal{R}$ small boxes validated to enclose a unique local minimum
  (in the unconstrained case), or a feasible point with small
  objective function value (in the constrained case); and
$\mathcal{U}$ small boxes not validated either to enclose a minimum
  or *not* to enclose a minimum.

Find enclosures for f^* and X^* such that $f^* = f(X^*) \le f(X)$, for all $X \in \boldsymbol{X}_0 \subset \mathbf{R}^n$:

```
Initialize L = {X_0}      // List of pending boxes
f̄ = +∞      // f* ≤ f̄
DO WHILE L is not empty
   Get current box X^(c) from L
   IF f̄ < f(X^(c)) THEN
      Discard X^(c)      // We can guarantee that there is no global minimum in X^(c)
   ELSE
      Attempt to improve f̄
   END IF
   IF we can otherwise guarantee that there is no global minimum in X^(c) THEN
      Discard X^(c)
   ELSE IF we can validate the existence of a unique local minimum in X^(c) THEN
      Shrink X^(c), and add it to R
   ELSE IF X^(c) is small THEN
      Add X^(c) to U
   ELSE
      Split X^(c) into sub-boxes, and add them to L
   END IF
END DO
```

The actual GlobSol code is much more sophisticated than the above suggests, but the concepts are similar. See [15] or [16] for details.

9.2. The Unique Features of GlobSol

Some features of GlobSol that distinguish it from other interval global optimization packages are briefly outlined here. See [15], [16], or more details at the World Wide Web page `www.mscs.mu.edu/~globsol/User_Guide/13WhatIs/`

- The objective function and constraints are specified as Fortran 90 programs.

- Globsol can be configured to use constraint propagation (substitution/iteration) on the intermediate quantities in objective and constraint function evaluation. For some objectives and constraints, this can result in significant speedup.
- GlobSol can be configured to use an overestimation-reducing "peeling" process for bound-constraints.
- GlobSol uses a novel and effective point method to find approximate feasible points. When an approximate feasible point is found, GlobSol verifies bounds within which a true feasible point must exist, from which upper bounds for the global optima may be obtained.
- GlobSol has a special augmented system mode for least squares problems.
- Globsol uses epsilon-inflation and set-complementation, with carefully controlled tolerances, to avoid singularity problems, and to facilitate verification.

9.3. An Illustrative Example of GlobSol's Use

This simple example (illustrative, not real-world) illustrates use of GlobSol. Suppose the optimization problem is

$$\text{minimize} \quad \phi(X) = -2x_1^2 - x_2^2$$

subject to constraints

$$\begin{aligned} x_1^2 + x_2^2 - 1 &\le 0 \\ x_1^2 - x_2 &\le 0 \\ x_1^2 - x_2^2 &= 0. \end{aligned}$$

The following Fortran 90 program communicates this objective function and constraints to GlobSol.

```
PROGRAM SIMPLE_MIXED_CONSTRAINTS
  USE CODELIST_CREATION
      PARAMETER (NN=2)
      PARAMETER (NSLACK=0)
      TYPE(CDLVAR), DIMENSION(NN+NSLACK):: X
      TYPE(CDLLHS), DIMENSION(1):: PHI
      TYPE(CDLINEQ), DIMENSION(2):: G
      TYPE(CDLEQ), DIMENSION(1) :: C

      OUTPUT_FILE_NAME='MIXED.CDL'
      CALL INITIALIZE_CODELIST(X)

  PHI(1) = -2*X(1)**2 - X(2)**2
  G(1) = X(1)**2 + X(2)**2 - 1
  G(2) = X(1)**2 - X(2)
  C(1) = X(1)**2 - X(2)**2
```

```
          CALL FINISH_CODELIST
     END PROGRAM SIMPLE_MIXED_CONSTRAINTS
```

A data file of the following form communicates the bounds $[0,1] \times [0,1]$ on the search region to GlobSol.

```
     1D-5        ! General domain tolerance
      0   1      ! Bounds on the first variable
      0   1      ! Bounds on the second variable
     F F         ! X(1) has no bound constraints
     F F         ! X(2) has no bound constraints
```

Separate configuration files supply algorithm options, such as which interval Newton method to use and how to precondition the linear systems.

Running the above program produces an internal representation, or code list. The optimization code then interprets the code list at run time to produce floating point and interval evaluations of the objective function, gradient, and Hessian matrix. An output file of the following form is then produced.

```
 Output from FIND_GLOBAL_MIN on  06/28/1998  at  16:28:09.
 Version for the system is: June 15, 1998

 Codelist file name is: MIXEDG.CDL
 Box data file name is: MIXED.DT1

 Initial box:
[    0.000000000000000000E+00,   0.100000000000000000E+01 ]
[    0.000000000000000000E+00,   0.100000000000000000E+01 ]

 BOUND_CONSTRAINT:
   F F   F F

 ---------------------------------------
 CONFIGURATION VALUES:

 EPS_DOMAIN:   0.1000E-04   MAXITR:   60000
 DO_INTERVAL_NEWTON: T  QUADRATIC: T  FULL_SPACE: F
 VERY_GOOD_INITIAL_GUESS:F
 USE_SUBSIT:T
 OUTPUT UNIT:7 PRINT_LENGTH:1
 Default point optimizer was used.

 THERE WERE NO BOXES IN COMPLETED_LIST.

 LIST OF BOXES CONTAINING VERIFIED FEASIBLE POINTS:

 Box no.:1
 Box coordinates:
[    0.707106766730664638E+00,   0.707106795645393915E+00 ]
[    0.707106766730642544E+00,   0.707106795645415898E+00 ]
```

```
PHI:
[   -0.150000006134372299E+01,  -0.149999993866885117E+01 ]
 Level: 3
 Box contains the following approximate root:
   0.707106781188029276E+00   0.707106781188029165E+00
 OBJECTIVE ENCLOSURE AT APPROXIMATE ROOT:
[   -0.150000000000628830E+01,  -0.150000000000628431E+01 ]
 Unknown = T   Contains_root =T
 U0:
[    0.385164798606970116E+00,   0.385164815661114435E+00 ]
 U:
[    0.577747199465034389E+00,   0.577747221937741084E+00 ]
[    0.000000000000000000E+00,   0.100000000000000000E+01 ]
 V:
[    0.192582392641276468E+00,   0.192582414496085957E+00 ]
 INEQ_CERT_FEASIBLE:
  F T
 NIN_POSS_BINDING:1

 ALGORITHM COMPLETED WITH LESS THAN THE MAXIMUM NUMBER,
       60000  OF BOXES.
 Number of bisections: 3
 No. dense interval residual evaluations -- gradient code list: 83
 Number of orig. system inverse midpoint preconditioner rows: 3
 Number of orig. system C-LP preconditioner rows: 109
 Number of Gauss--Seidel steps on the dense system: 112
 Number of gradient evaluations from a gradient code list: 13
 Total number of dense slope matrix evaluations: 55
 Number of times the interval Newton method made a coordinate
       interval smaller: 60
 Number of times a box was rejected because the constraints were
       not satisfied: 1
 Total time spent doing linear algebra (preconditioners
       and solution processes): 0.15018546581268311
 Number of times the approximate solver was called: 2
 Number Fritz-John matrix evaluations: 19
 Number times SUBSIT decreased one or more
       coordinate widths: 1
 Number times a box was rejected due infeasible
       inequality constraints: 2
 BEST_ESTIMATE:   -0.1500E+01
 Total number of boxes processed in loop: 7
 Overall CPU time:    0.2277E+00
 CPU time in PEEL_BOUNDARY:    0.1365E-03
 CPU time in REDUCED_INTERVAL_NEWTON:    0.1458E+00
```

This report says that GlobSol guarantees there is a minimizer satisfying the constraints in the box $[0.\underline{7071067}66, 0.\underline{7071067}96] \times [0.\underline{7071067}66, 0.\underline{7071067}96]$. The minimum feasible value of the objective is in the interval $[-1.\underline{5000000}62, -1.\underline{4999999}37]$. The intervals computed by GlobSol enclose the true values $X^* = (1/\sqrt{2}, 1/\sqrt{2})$ and $f^* = -3/2$. GlobSol required 83 function, and 13 gradient evaluations, and 0.23 CPU seconds on a Sun SPARC Ultra 140.

10. Conclusions

We have applied validated global optimization techniques to several practical industrial applications. For each application, we have successfully solved at least modest prototypical problems and have identified modeling and algorithmic advances necessary for success on problems whose size and complexity is of genuine industrial importance. Work is continuing on several of these applications to improve our models and to improve and parallelize the algorithms used by GlobSol.

References

1. Bill P. Buckles and Fred Petry. *Genetic Algorithms.* IEEE Computer Society Press, 1992.
2. Paulina Chin, Robert M. Corless, and George F. Corliss. Globsol case study: Inexact greatest common denominators. Technical Report, Marquette University Department of Mathematics, Statistics, and Computer Science, Milwaukee, Wisc., 1998. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn_gcd1.ps`.
3. Paulina Chin, Robert M. Corless, and George F. Corliss. Optimization strategies for the floating-point GCD. In *ISSAC Proceedings*, page to appear, 1998. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/gcd.ps`.
4. Robert M. Corless. Cofactor iteration. *SIGSAM Bulletin: Communications in Computer Algebra*, 30(1):34–38, March 1996.
5. George F. Corliss, Chenyi Hu, R. Baker Kearfott, and G. William Walster. Rigorous global search – Executive summary. Technical Report No. 442, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., April 1997. Global Solutions Working Note 1. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn01.ps`.
6. Joseph Daniels, Shirish Ranjit, R. Baker Kearfott, and George F. Corliss. Globsol case study: Currency trading (Swiss Bank Corp.). Technical Report, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., 1998. Global Solutions Working Note 7. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn07.ps`.
7. Xin Feng, Rudi Yang, Yonghe Yan, Yunchuan Zhu, George F. Corliss, and R. Baker Kearfott. Globsol case study: Parameter optimization for the eddy current compensation of MRI coils (General Electric Medical). Technical Report, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., 1998. Global Solutions Working Note 8. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn08.ps`.
8. Frank Fritz. Validated global optimization and finite element analysis. Master's thesis, Marquette University Department of Mathematics, Statistics, and Computer Science, Milwaukee, Wisc., November 1998.
9. Frank Fritz, George F. Corliss, Andrew Johnson, Donald Prohaska, and Jonathan Hart. Globsol case study: Rocket nozzle design (MacNeal-Schwindler). Technical Report, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., 1998. Global Solutions Working Note 5. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn05.ps`.
10. Elden R. Hansen. *Global Optimization Using Interval Analysis.* Marcel Dekker, New York, 1992.
11. R. Horst and M. Pardalos. *Handbook of Global Optimization.* Kluwer, Dordrecht, Netherlands, 1995.

12. Ron Van Iwaarden. *An Improved Unconstrained Global Optimization Algorithm.* PhD thesis, University of Colorado at Denver, Denver, Colorado, 1996. Electronic copy: `http://www-math.cudenver.edu/~rvan/phd.ps.gz`.
13. Christian Jansson. A global optimization method using interval arithmetic. *IMACS Annals of Computing and Applied Mathematics*, 1992.
14. Christian Jansson. On self-validating methods for optimization problems. In J. Herzberger, editor, *Topics in Validated Computations*, pages 381–439, Amsterdam, Netherlands, 1994. North-Holland.
15. R. Baker Kearfott. *Rigorous Global Search: Continuous Problems.* Kluwer Academic Publishers, Dordrecht, Netherlands, 1996.
16. R. Baker Kearfott, M. Dawande, K. S. Du, and Chenyi Hu. Algorithm 737: INTLIB: A portable Fortran 77 interval standard library. *ACM Trans. Math. Software*, 20(4):447–458, 1994.
17. Richard Kelnhoffer. *Applications of Interval Methods to Parametric Set Estimation from Bounded Error Data.* PhD thesis, Marquette University Department of Electrical and Computer Engineering, 1997. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/kelnhoffer.ps`.
18. Arnold Neumaier. *Interval Methods for Systems of Equations.* Cambridge University Press, Cambridge, 1990.
19. Shirish Ranjit. Risk management of currency portfolios. Master's thesis, Marquette University Department of Economics, Milwaukee, Wisc., April 1998. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/ranjit.ps`.
20. Dietmar Ratz. An inclusion algorithm for global optimization in a portable PASCAL-XSC implementation. In L. Atanassova and J. Herzberger, editors, *Computer Arithmetic and Enclosure Methods*, pages 329–338, Amsterdam, Netherlands, 1992. North-Holland.
21. Dietmar Ratz. Box-splitting strategies for the interval Gauss–Seidel step in a global optimization method. *Computing*, 53:337–354, 1994.
22. Dietmar Ratz and Tibor Csendes. On the selection of subdivision directions in interval branch-and-bound methods for global optimization. *Journal of Global Optimization*, 7:183–207, 1995.
23. Paul J. Thalacker, Kristie Julien, Peter G. Toumanoff, Joseph P. Daniels, George F. Corliss, and R. Baker Kearfott. Globsol case study: Portfolio management (Banc One). Technical Report, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., 1998. Global Solutions Working Note 6. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn06.ps`.
24. Zhitao Wang, Peter Tonellato, George F. Corliss, and R. Baker Kearfott. Globsol case study: Gene prediction (Genome Therapeutics). Technical Report, Department of Mathematics, Statistics and Computer Science, Marquette University, Milwaukee, Wisc., 1998. Global Solutions Working Note 4. Electronic copy: `http://www.mscs.mu.edu/~globsol/public_html/Papers/wn04.ps`.