===
This survey appeared in Optima of Dec 1996, pp. 1-8
===

CONTINUOUS GLOBAL OPTIMIZATION SOFTWARE: A BRIEF REVIEW

J\'{a}nos D. Pint\'{e}r
Pint\'{e}r Consulting Services (PCS), and Dalhousie University
PCS address: 129 Glenforest Drive, Halifax, NS, Canada B3M 1J2
e-mail: pinter@tuns.ca http://www.tuns.ca/~pinter/

Abstract

Following a concise introduction to multiextremal mathematical programming problems and global optimization (GO) strategies, a commented list of software products for analyzing and solving continuous GO problems is presented.

Keywords:

Multiextremal optimization models; continuous global optimization; solution approaches; software review.

AMS Subject Classification: 65K30, 90C05.

1. Global Optimization Models and Solution Approaches

A large variety of quantitative decision issues, arising in the sciences, engineering and economics, can be perceived and modelled as a constrained optimization problem. According to this generic description, the best decision - often expressed by a real vector - is sought which satisfies all stated feasibility constraints, and minimizes (or maximizes) the value of an objective function. Applying standard mathematical programming notation, we shall consider problems in the general form

(1) min f(x) subject to x \epsilon D \in R^n.

The function f symbolizes the objective(s) in the decision problem, and D denotes the (non-empty) set of feasible decisions. D is usually defined by a finite number of functions; for the purposes of the present discussion, we shall assume that

(2) D={x \epsilonD \in R^n: l \le x \le u; g_j(x) \le 0 j=1,...,J}.

In (2) l and u are explicit (finite) bounds, and g_j j=1,...,J are given constraint functions. Postulating now that all functions defined above are continuous, the optimal solution set to problem (1)-(2) is non-empty.

Most typically, it is assumed that the decision problem modelled by (1)-(2) has a unique - locally and, at the same time, also globally - optimal solution. Uniextremality is often implied by the mathematical model structure (for example, by the strict convexity of f, and the convexity of D). This paradigm corresponds to the situation in which one, supposedly, has a sufficiently close initial 'guess' of the feasible region where the optimal solution x^* is located. Hence, the global optimality of the solution directly follows, having found the single local optimum of f on D. For example, linear and convex nonlinear programming models - both, in essence, satisfying the mentioned uniextremality assumption in most practical cases - have been extensively applied in the past decades to formulate and solve an impressive range of decision problems.

Although very important classes of models naturally belong to the above category, there is also a broad variety of problems, in which the property of uniextremality cannot be simply postulated or verified. Consider, for instance, the following general problem types:

\bullet nonlinear approximation, including the solution of systems of nonlinear equations and inequalities
\bullet model fitting to empirical data (calibration, parameterization)
\bullet optimized design and operation of complex 'black box' ('oracle') systems, e.g., in diverse engineering contexts
\bullet configuration/arrangement design (e.g., in various data classification, facility location, resource allocation, or scientific modelling contexts)

Such problems - together with numerous other prospective application areas - are discussed by Pint\'{e}r (1996), and the extensive list of related references therein. For further applications consult, e.g., Pardalos and Rosen (1987), T\"{o}rn and \v{Z}ilinskas (1989), Floudas and Pardalos (1990, 1992), Grossmann (1996), Bomze, Csendes, Horst and Pardalos (1996), or special application-related issues of the Journal of Global Optimization.

The emerging field of Global Optimization (GO) deals with mathematical programming problems, in the (possible) presence of multiple local optima. Observe that, typically, the number of local (pseudo)solutions is unknown and it can be quite large. Furthermore, the quality of the various local and global solutions may differ significantly. In presence of such structure - often visualized by 'hilly landscapes' corresponding to projections of the objective function into selected subspaces (given by coordinate-pairs of the decision variable x) - GO problems can be extremely difficult. Hence, most classical numerical approaches are, generally speaking, not directly applicable to solve them. For illustration, see Figure 1 which displays a - relatively simple - composition of trigonometric functions with imbedded polynomial arguments, in just two variables (denoted by x and y).

Figure 1 follows here

Caption: Figure 1. A multiextremal function in two variables

Naturally, under such circumstances, it is essential to use a proper global search strategy. Furthermore, instead of 'exact' solutions, most typically one has to accept diverse numerical approximations to the globally optimal solution (set) and optimum value.

Following early sporadic work related to GO (since the late fifties), the present state-of-art is characterized by several tens of monographs, a professional journal and - at least - a few thousand research articles devoted primarily to the subject. A few illustrative references are provided at the end of this brief review.

The most important GO model-classes which have been extensively studied include the following examples. (Please recall the general model form (1)-(2), and note that the problem-classes listed below are not necessarily distinct: in fact, several of them are hierarchically contained by more general problem-types listed.)

\bullet Bilinear and biconvex programming (f is bilinear or biconvex, D is convex)
\bullet Combinatorial optimization (problems which have discrete decision variables in f and/or in g_j j=1,...,J can be equivalently reformulated as GO problems in continuous variables)
\bullet Concave minimization (f is concave, D is convex)
\bullet Continuous global optimization (f is continuous, D is compact)
\bullet Differential convex (D.C.) optimization (f and g_j j=1,...,J can all be represented, as the difference of two corresponding convex functions)
\bullet Fractional programming (f is the ratio of two real functions, and g_j j=1,...,J are convex)
\bullet Linear and nonlinear complementarity problems (f is the scalar product of two vector functions, D is typically convex)
\bullet Lipschitz optimization (f and g_j j=1,...,J are arbitrary Lipschitz-continuous functions)
\bullet Minimax problems (f is some minimax objective, the maximum is considered over a discrete set or a convex set, D is convex)
\bullet Multilevel optimization (models non-cooperative games, involving hierarchies of decision-makers, their conflicting criteria are aggregated by f; D is typically assumed to be convex)
\bullet Multiobjective programming (e.g., when several conflicting linear objectives are to be optimized over a polyhedron)
\bullet Multiplicative programming (f is the product of several convex functions, and g_j j=1,...,J are convex, or - more generally - multiplicative functions)
\bullet Network problems (f can be taken from several function-classes including nonconvex ones, and g_j j=1,...,J are typically linear or convex)
\bullet Parametric nonconvex programming (the feasible region D and/or the objective f may also depend on a parameter vector)
\bullet Quadratic optimization (f is an arbitrary - indefinite - quadratic function; g_j j=1,...,J are linear or, in the more general case, can be arbitrary quadratic functions)
\bullet Reverse convex programming (at least one of the functions g_j expresses a reverse convex constraint)
\bullet Separable global optimization (f is an arbitrary nonlinear - in general, nonconvex - separable function, D is typically convex)
\bullet Various other nonlinear programming problems, including, e.g., nonconvex stochastic models (in which the defining functions f, g_j j=1,...,J depend on random factors, possibly in an implicit, 'black box' manner)

For detailed descriptions of most of these model-types and their connections consult, e.g., Horst and Pardalos (1995), with numerous further references.

There are several main classes of algorithmic GO approaches which possess strong theoretical convergence properties, and - at least in principle - are straightforward to implement and apply. All such rigorous GO approaches have an inherent computational demand which increases non-polynomially, as a function of problem-size, even in the simplest GO instances. It should be emphasized at this point that GO approaches are (should be) typically completed by a 'traditional' local optimization phase - at least when considering also numerical efficiency issues. Global convergence, however, needs to be guaranteed by the global scope algorithm-component which - theoretically - should be used in a complete, 'exhaustive' fashion. These remarks indicate the significant difficulty of developing robust and efficient GO software.

Without aiming at completeness, several of the most important GO strategies are listed below; for details, consult, for instance, the corresponding works from the list of references. (Note that the listing below is not complete, and its items are not necessarily mutally exclusive: some software implementations combine ideas from several approaches.)

\bullet Adaptive partition and search strategies (including, i.a., branch-and-bound algorithms, Bayesian approaches and interval arithmetic based methods) (Forg\'{o}, 1988; Ratschek and Rokne, 1988; Mockus, 1989; Neumaier, 1990; Zhigljavsky, 1991; Hansen, 1992; Horst and Pardalos, 1995; Horst and Tuy, 1996; Pint\'{e}r, 1996; Kearfott, 1996)

\bullet Adaptive stochastic search algorithms (including random search, simulated annealing, evolution and genetic algorithms) (van Laarhoven and Aarts, 1987; Zhigljavsky, 1991; Horst and Pardalos, 1995; Michalewicz, 1996; Pint\'{e}r, 1996)

\bullet Enumerative strategies (for solving combinatorial problems, or certain 'structured' - e.g., concave - optimization problems) (Forg\'{o}, 1988; Horst and Pardalos, 1995; Horst and Tuy, 1996)

\bullet 'Globalized' local search methods (applying a grid search or random search type global phase, and a local search algorithm) (Horst and Pardalos, 1995; Pint\'{e}r, 1996)

\bullet Heuristic strategies (deflation, tunneling, filled function methods, approximate convex global underestimation, tabu search, etc.) (Horst and Pardalos, 1995; Pint\'{e}r, 1996)

\bullet Homotopy (parameter continuation) methods and related approaches (including fixed point methods, pivoting algorithms, etc.) (Horst and Pardalos, 1995)

\bullet Passive (simultaneous) strategies (uniform grid search, pure random search) (Zhigljavsky, 1991; Horst and Pardalos, 1995; Pint\'{e}r, 1996)

\bullet Successive approximation (relaxation) methods (cutting plane, more general cuts, minorant construction approaches, certain nested optimization and decomposition strategies) (Forg\'{o}, 1988; Horst and Pardalos, 1995; Pint\'{e}r, 1996)

\bullet Trajectory methods (differential equation model based, path-following search strategies) (Horst and Pardalos, 1995)

In spite of a considerable progress related to the rigorous theoretical foundations of GO, software development and 'standardized' use lag behind. The main reason for this is, of course, the inherent numerical difficulty of GO, even in case of 'simpler' specific instances (such as, e.g., the indefinite quadratic programming problem). In general, the difficulty of a global optimization problem (GOP) can be expected to increase as some exponential function of the problem dimension n. Consequently, dimensions 100, 50 or even 10 can already be considered as 'large', depending on the GOP type investigated (please recall Figure 1). In the remainder of this paper, an illustrative list of software products to solve GOPs is reviewed.

2. GO Software: Information Sources and Some General Remarks

For the purposes of collecting information to this survey, GO software authors have been asked (mainly by sending e-mail messages, and by placing 'electronic ads' at several prominent mathematical programming sites on the WWW) to submit documentation related to their work. The information - or lack thereof - summarized below is largely based on the responses received. Additional information has been collected from the Internet, from several GO books, and from the Journal of Global Optimization. Note that though in many research publications reference is made to numerical examples, or even to sophisticated specific applications, only such work is reported below which is understood to be a general purpose, 'off-the-shelf' - i.e., at least in principle readily available, and legally distributable - program system for handling a particular GOP class.

For obvious reasons, the present survey is far from being 'complete' in any possible sense; rather, it is an attempt to provide a realistic picture of the state-of-art, supported by instances of existing software. This short review is not intended to be either comparative or 'judgemental': one simple reason being that the information received from GO software developers is used 'as is', mostly without the possibility of actual software testing. By the same token, the accuracy of all information cannot be guaranteed, either. Further research into this direction - including the preparation of a more comprehensive and detailed survey - is thought to be important; some of that work is currently in progress.

The software list provided in the next section is simply alphabetical, without categorization (which - in absence of detailed information - would possibly be imprecise anyway). For a more uniform presentation style, abbreviations are associated with all software products listed, even when such names were not given in the documentation available for this survey (existing names were not changed, of course). The descriptions are almost formula-free and extremely concise - due to space restrictions. For the latter reason, we decided not to include important classes of more specific GO approaches and related methodology. In particular - as reflected by the title - pure or mixed integer programming, and more general combinatorial optimization algorithms are not discussed here. Furthermore, although most of the available top-of-the-line continuous nonlinear (convex) optimization software can be applied - with good taste and some luck - to analyze GOPs, even the most prominent such systems are excluded from this review. Again, it is planned to prepare a more detailed survey, appropriately discussing also the program system types mentioned.

The hardware and software platform of the systems reviewed is also shown, when such information is available. In order to assist the Reader in obtaining additional information, contact person(s), their e-mail addresses, ftp and/or WWW sites are listed, whenever known to me. (For brevity, only a few such pointers are provided in each case.)

The Reader is assumed to have at least some basic familiarity with the GO approaches mentioned: for related discussions, please consult the references.

3. Short Software Descriptions

\alphaBB -- A GO Algorithm for General Nonconvex Problems

An implementation of a Branch-and-Bound (B&B) algorithm which is based on the difference of convex functions (D.C.) transformation. Nonconvexities are identified and categorized as of either special or generic structure. Special nonconvex (such as bilinear or univariate concave) terms are convex lower bounded using customized bounding functions. For generic nonconvex terms, convex lower bounding functions are derived by utilizing the parameter \alpha (specified by the user, or derived based on theory). \alphaBB solves general unconstrained and constrained problems; it requires MINOS and/or NPSOL for the solution of linear or convex optimization subproblems. (Languages: C and Fortran.) Contact: I.P. Androulakis <androula@titan.princeton.edu>, C.A. Floudas <floudas@titan.princeton.edu>, C.D. Maranas <kmaranas@titan.princeton.edu>, http://titan.princeton.edu/.

ANNEAL - Simulated Annealing

ANNEAL is based on the core SA approach, including several possibilities of parameter adjustment, and a deterministic solution refinement phase. It has been applied to predict complex crystal structures. Workstation implementation. Contact: W. Bollweg <bollweg@math.uni-muenster.de>, H. Maurer <maurer@math.uni-muenster.de>, H. Kroll <kroll@uni-muenster.de>.

ASA - CalTech Adaptive Simulated Annealing

ASA is developed to find the global optimum of a continuous non-convex function over a multidimensional interval (box). This algorithm permits an annealing schedule for 'temperature' decreasing exponentially in annealing time. The introduction of re-annealing also permits adaptation to changing sensitivities in the parameter-space. Some other adaptive options in ASA include self-optimize (to find optimal starting conditions) and quenching (to methodically find faster performance that might be useful for large parameter-spaces). (Language: C.) Contact: L. Ingber <ingber@alumni.caltech.edu>, http://www.ingber.com/#ASA-CODE.

B&B - A Family of B&B Algorithms

This obvious acronym (by the present author) attempts to summarize several B&B type algorithms developed to solve certain structured GOP classes. These include (among others) indefinite quadratic, quasiconvex-concave, and general Lipschitz problems. Workstation implementations. (Language: C.) Contact: R. Horst <horst@orsun3.uni-trier.de>, M. Nast <nast@orsun1.uni-trier.de>, N. Thoai <thoai@orsun2.uni-trier.de>.

BARON - Branch-And-Reduce Optimization Navigator

Combines interval analysis and duality with enhanced B&B concepts. The BARON modules can handle structured nonconvex problems up to thousands of constraints and variables. The library of specialized modules includes solvers for numerous specific GOP-classes. (For other, more general problems, underestimation routines need to be provided by the user.) All modules can solve also such problems in which some or all of the variables are restricted to integer values. The specialized modules use OSL or MINOS, to solve interim subproblems. Workstations, UNIX type operating systems.

(Languages: Fortran and GAMS.) Contact: N.V. Sahinidis, http://archimedes.me.uiuc.edu/sigma/baron.html, ftp://aristotle.uiuc.edu.

BGO - Bayesian Global Optimization

This program system includes four versions of Bayesian search, clustering, uniform deterministic grid, and pure Monte Carlo search. Bound constraints and more general constraints can be handled. Interactive DOS and UNIX versions are available. (Languages: Fortran and C.) Contact: J. Mockus <jonas.mockus@mii.lt>, L. Mockus <mockus@ecn.purdue.edu>.

cGOP - Global Optimization Program

Solves structured GOPs which have an objective function of the form $a^Tx+b^Ty+x^TAy+f_1(x)+f_2(y)$ with convex f_1, f_2, and linear constraints. Requires the presence of the commercial codes MINOS and/or CPLEX to solve linear, mixed-integer linear and convex subproblems. cGOP has been used to solve problems involving several hundred variables and constraints. Versions are available for workstations. (Language: C.) Contact: V. Visweswaran <vishy@titan.princeton.edu>, C.A. Floudas <floudas@titan.princeton.edu>, http://titan.princeton.edu/.

CGU - Convex Global Underestimator

This approach is designed to generate efficient approximations to the global minimum of a multiextremal function, by fitting a convex function to the set of all known (calculated) local minima. This heuristically attractive strategy requires only the sequential solution of auxiliary LPs, and some rather elementary calculations. CGU has been applied to calculate molecular structure predictions, up to several tens of variables. Implemented on parallel workstations and supercomputers.
Contact: K.A. Dill, A.T. Phillips <phillips@haney.scs.usna.navy.mil>, J.B. Rosen <rosen@cs.umn.edu>.

CRS - Controlled Random Search

This is a recently developed variant of a popular class of random search based methods which can be applied under very mild analytical conditions imposed on the GOP. Several other related stochastic search methods have also been developed by this group. Workstation implementations.
Contact: M.M. Ali, A. T\"{o}rn <torn@finabo.abo.fi>, S. Viitanen <sami.viitanen@abo.fi>.

CURVI - Bound-Constrained Global Optimization

Windward Technologies (WTI) develops advanced numerical and visualization software, for solving constrained and unconstrained nonlinear optimization problems. One of their solvers, CURVI is aimed at solving bound-constrained nonlinear programs which have a complicated - possibly multiextremal - objective function. (Language: Fortran.) Contact: T. Aird <tomaird@aol.com>, http://users.aol.com/WTI/.

DE - Differential Evolution Genetic Algorithm for Bound Constrained GO

DE won the third place at the 1st International Contest on Evolutionary Computation on a real-valued function test-suite. It was the best genetic algorithm approach (the first two places of the contest were won by non-GA algorithms). (Languages: Matlab and C.) Contact: R. Storn <storn@icsi.berkeley.edu>, http://http.icsi.berkeley.edu/~storn/code.html.

ESA - Edge Searching Algorithm

An implementation of an edge search algorithm for finding the global solution of linear reverse convex programs. ESA is based on an efficient search technique and the use of fathoming criteria on the edges of the polytope representing the linear constraints. In addition, the method incorporates several heuristics, including a cutting plane technique which improves the overall performance. Implemented for several UNIX platforms; the TPG Test Problem Generator is also available. (Language: Fortran.) Contact: K. Moshirvaziri <moshir@beach.csulb.edu>, <moshir@ee.ucla.edu>.

GA - Genetic Algorithms

Genetic algorithms - as a rule - can be applied to GOPs under mild structural requirements. Both general and specific information related to this popular solver class is available from the following sources:
- A Commented List of Genetic Algorithm Codes,
  ftp://ftp.germany.eu.net/pub/research/softcomp/ec/faq/www/q20_1.htm
- GA Archive, http://www.aic.nrl.navy.mil/galist/src/.

Only a few illustrative examples are listed in the present review.

GAS - Genetic Algorithm

Unconstrained and bound-constrained versions are available. For DOS and UNIX operating systems. (Language: C++.) Contact: M. Jelasity <jelasity@inf.u-szeged.hu), J. Dombi <dombi@inf.u-szeged.hu>, ftp://ftp.jate.u-szeged.hu/pub/math/optimization/GAS/.

GAucsd - Genetic Algorithm

Developed and maintained at the University of California, San Diego. GAucsd was written in C under Unix, but should be easy to port to other platforms. The package is accompanied by brief information and a User's Guide. (Language: C.) Contact: nici@ucsd.edu, GAucsd-request@cs.ucsd.edu, ftp://cs.ucsd.edu/pub/GAucsd/.

GENERATOR - Genetic Algorithm Solver

This method is aimed at solving a variety of (combinatorial and continuous multiextremal) scientific and engineering optimization problems. It is designed to interact with Excel which serves as a user interface. (Platform: Excel.) Contact: New Light Industries <nli@comtch.iea.com>, http://www.iea.com/~nli/.

GC - Global Continuation

GC is a continuation approach to GO applying global smoothing, in order to derive a simpler approximation to the original objective function. GC is applied by the authors to distance geometry problems, in the context of molecular chemistry modelling. IBM SP parallel system implementation. Contact: J.J.Mor\'{e} <more@mcs.anl.gov>, Z. Wu.

GENOCOP III - Genetic Algorithm for Constrained Problems

Solves general GOPs, in presence of additional constraints and bounds (using quadratic penalty terms). System parameters, domains, and linear inequalities are input via a data file. The objective function and any nonlinear constraints are to be given in appropriate C files. (Language: C.) Contact: Z. Michalewicz, http://www.coe.uncc.edu/~zbyszek/gcreadme.html, ftp://ftp.uncc.edu/coe/evol/genocopIII.tar.Z.

GEODES - Minimum-Length Geodesic Computing

Approximating a minimum-length geodesic on a multidimensional manifold, GEODES is differential geometry software. However, it has potential also in the GO context. GEODES includes example manifolds and metrics; it is implemented in Elements (a matrix and function oriented scientific modelling environment) to compute and visualize geodesics on 2D surfaces plotted in 3-space. Portable to various hardware platforms. (Languages: C, C++.) Contact: W.L. Anderson <elements@ix.netcom.com>, http://www.netcom.com/~elements/, http://www.netlib.org/ode/geodesic/.

GLO - Global and Local Optimizer

GLO is a modular optimization system developed for 'black box' problems in which objective function calculations may take a long time. Its methodology is based on the coupling of global (genetic) and local (variable metric) nonlinear optimization software with scientific applications software. It has been applied to automated engineering design. Besides the modular optimization control system, GLO also has a graphical user interface, and includes a pre-processor. Contact: M.J. Murphy, http://www.llnl.gov/glo/09glo.html, M. Brosius <brosius2@llnl.gov>.

GLOBAL - Multistart with Stochastic Clustering

GLOBAL can be used for the solution of the general bound-constrained GOP which has a (measurable) real objective function. The algorithm is a derivative-free implementation of the clustering stochastic multistart method of Boender et al., supplemented with a quasi-Newton local search routine and with a robust random local search method. Available for UNIX machines, IBM-compatible mainframes and PCs. (Languages: Fortran and C.) Contact: T. Csendes <csendes@inf.u-szeged.hu>, http://www.inf.u-szeged.hu/~csendes/, ftp://ftp.jate.u-szeged.hu/pub/math/optimization/index.html.

GLOBALIZER - An Educational Program System for Global Optimization

Serves for solving univariate GOPs. After stating the problem, the user can choose among various (random search, B&B based, or Bayesian partition based) solver techniques. The software has interactive tutoring capabilities, provides textual and graphical information. Works on PCs, under MS-DOS. Contact: R.G. Strongin <strongin@nnucnit.nnov.su>, V.P. Gergel, A.V. Tropichev.

GLOPT - Constrained Global Optimization

Solves GOPs with a block-separable objective function subject to bound constraints and block-separable constraints: it finds a nearly globally optimal point that is near to a true local minimizer. GLOPT uses a B&B technique to split the problem recursively into subproblems that are either eliminated or reduced in their size. It includes a new reduction technique for boxes and new ways for generating feasible points of constrained nonlinear programs. The current implementation of GLOPT uses neither derivatives nor simultaneous information about several constraints. (Language: Fortran.) Contact: A. Neumaier <neum@cma.univie.ac.at>, S. Dallwig and H. Schichl.

GOPP - Global Optimization of Polynomial Problems using Gr\"{o}bner Bases

The (local) optimality conditions to polynomial optimization problems lead to polynomial equations, under inequality constraints. Applying recent Gr\"{o}bner basis techniques, this approach is aimed at finding all solutions to such systems, hence also finding global optima. (Language:

Maple.) Contact: K. Hagglof <kroffa@math.kth.se>, P.O. Lindberg <pol@math.kth.se>, L. Svensson <larss@math.kth.se>, http://www.optsyst.math.kth.se.

GOT - Global Optimization Toolbox

GOT combines random search and local (convex) optimization. DOS and HP UX versions are available. (Language: Fortran.)
Contact: A.V. Kuntsevich <kun@d120.icyb.kiev.ua>.

GSA - Generalized Simulated Annealing

GSA is based on the generalized entropy by Tsallis. The algorithm obeys detailed balance conditions and, at low 'temperatures', it reduces to steepest descent. (Note that the members of the same research group have been involved in the development of several SA type algorithms.) Contact: J.E. Straub <straub@chem.bu.edu>, P.Amara <amara@chem.bu.edu>, J. Ma <jma@chem.bu.edu>.

IHR - Improving Hit-and-Run

IHR is a random search based GO algorithm that can be used to solve both continuous and discrete optimization problems. IHR generates random points in the search domain by choosing a random direction and selecting a point in that direction. Versions have been implemented, using different distributions for the random direction, as well as several ways to randomly select points along the search line. The algorithm can also handle inequality constraints, and a hierarchy of objective functions. IHR has been used to solve GOPs in various disciplines such as in engineering design.
Contact: Z. Zabinsky <zelda@u.washington.edu>, ftp://ftp.bart.ieng.washington.edu.

IMINBIS - Interval Arithmetic Based GO

This method applies interval arithmetic techniques to isolate the stationary points of the objective function. Next, a topological characterization is used to separate minima from maxima and saddle points, followed by local minimization (sub)searches to select the global solution. The method has been applied also to 'noisy' problems. Workstation and PC implementations, extensive related research. (Language: Fortran.)
Contact: M.N. Vrahatis <vrahatis@mars.math.upatras.gr>, D.G. Sotiropoulos <dgs,@math.upatras.gr>, E.C. Triantaphyllou.

INTBIS - Global Solver for Polynomial Systems of Equations

Finds all solutions of polynomial systems of equations, with rigorously guaranteed results. The software package INTBIS is ACM-TOMS Algorithm 681; it is available through NETLIB. Distributed with the package are four source code files, sample input and output files, and a brief documentation file. The source files consist of the following: interval arithmetic, stack management, core INTBIS routines, and machine constantse. (Language: Fortran.) Contact: R.B. Kearfott <rbk@usl.edu>, http://interval.usl.edu/kearfott.html, ftp://interval.usl.edu/pub/interval_math/intbis/.

INTOPT_90 - Verified (Interval) Global Optimization

Serves to the verified solution of nonlinear systems of equations, and unconstrained and bound-and-equality-constrained global optimization. Based on exhaustive search, driven by a local optimizer, epsilon-inflation, interval Newton methods, and interval exclusion principles; uses automatic differentiation. Test results with hundreds of test examples. The underlying interval arithmetic package (ACM TOMS Algorithm 737) is also distributed. Workstation and PC implementations. (Language: Fortran.) Contact: R.B. Kearfott <rbk@usl.edu>, http://interval.usl.edu/kearfott.html, ftp://interval.usl.edu/pub/interval_math/intbis/.

INTGLO, INTGLOB - Integral Global Optimization

These methods solve unconstrained and constrained, as well as discrete GOPs by the integral method. They also include a discontinuous penalty function approach for constrained problems. Problems up to one hundred variables have been solved. A set of test problems is also available, including box or unconstrained, constrained, concave minimization, discrete variable programs and multicriteria programs. For IBM PCs. (Language: Fortran.) Contact: Q. Zheng <qzheng@linden.msvu.ca>, D. Zhuang <deming.zhuang@msvu.ca>.

ISA - Inductive Search Algorithm

ISA won first place at the 1st International Contest in Evolutionary Computation, on a real-valued function test-suite. (Language: C++.) Contact: G. Bilchev, information available at http://solon.cma.univie.ac.at/~neum/glopt/test_results.html#bilchev.

LGO - Continuous and Lipschitz Optimization

Solves bound-constrained and more general GOPs under mild structural requirements; it can be applied also to 'black box' problems. LGO integrates several global (adaptive partition and random search based) and local (derivative-free conjugate directions type) strategies: these can be activated in interactive or automatic execution modes. The PC version has a menu interface to assist the application development process, includes a concise information / tutoring session, and has visualization capabilities. Available also for workstations. LGO has been applied to problems with up to 100 variables (can be configured to encompass larger sizes). Accompanied by a User's Guide and sample problems. (Language: Fortran.) Contact: J.D. Pint\'{e}r <pinter@tuns.ca>, http://www.tuns.ca/~pinter/.

LOPS - Lipschitz Optimization Program System

In all approaches listed below, the objective function is defined over n-intervals. The Lipschitz-continuity of f or f' is also assumed. Problem-classes and corresponding available versions include:
- one-dimensional GOPs (sequential methods with local tuning, PC version (Language: C++)
- one-dimensional GOPs, parallel solver implementations (Language: Alliant FX/80, parallel Fortran)
- multi-dimensional GOPs: sequential and parallel algorithms using Peano curves (Language: Alliant FX/80, parallel Fortran)

Contact: Y.D. Sergeyev <yaro@si.deis.unical.it>.

MAGESTIC - Data Fitting by Global Optimization

Automatic global optimization based on a fast modified Gauss-Newton approach combined with Monte Carlo search. MAGESTIC handles calibration model variants (e.g., parameter and error masks for restricted sub-fitting, implicit equation fitting without solving, etc.). Suitable for use also with Lagrange multipliers for constrained optimization. Uses Excel as an interface (under Windows) and for generating graphics. (Platform: Excel.) Contact: Logix Consulting <magestic@lgx.com>, http://www.lgx.com/magestic.html.

MULTISTART - Clustering Algorithm

This widely used approach is based on random search - or some other initial sampling in the feasible set - combined with clustering, and local optimization launched from the most 'promising' point(s). Implemented on SUN workstations. Several interesting applications - in combination with simulation models - are related to the analysis of oil resources. (Language: Fortran.) Contact: S. Buitrago <saul.intevep.pdv.com>.

NETSPEAK - General Network Optimization

This is an algebraic modelling language used to specify, solve, and analyze general - linear, but also possibly nonconvex - minimum cost network flow problems. A wide variety of network and network-related topologies (pure networks, networks with side-constraints and/or side variables, generalized networks) can be modelled using NETSPEAK. The language is being developed as a Windows application; it features flexible I/O, robust program control, and intuitive commands. Contact: B.W. Lamar <bwlamar@canterbury.ac.nz>.

PA - Packet Annealing

In PA, the Gibbs distribution of the objective function is deterministically 'annealed' by tracing the evolution of a multiple Gaussian packet approximation. The approach has been applied to analyze complex molecular conformation models. IBM PC implementation. Contact: D. Shalloway <dis2@cornell.edu>, B.W. Church <bwc1@cornell.edu>, M. Oresic <matej@lancelot.bio.cornell.edu>.

PROFIL - Interval Branch and Bound Method

Bound constrained interval global optimization, with rigorously guaranteed results. PROFIL is based on BIAS (Basic Interval Arithmetic Subroutines) which provides an interface for interval operations. For PC and a number of UNIX systems. (Language: C.) Contact: C. Jansson, O. Knueppel <knueppel@tu-harburg.d400.de>,
http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html,
ftp://ti3sun.ti3.tu-harburg.de/pub/profil/unix/profopt.tar.Z,
ftp://ti3sun.ti3.tu-harburg.de/pub/profil/pc/profopt.tgz.

PVGO - Parallel Verified Global Optimization

PVGO is a new parallel method for interval global optimization. Implemented on a Connection Machine CM5. (Language: Pascal-XSC.) Contact: S. Berner <sonja@math.uni.wuppertal.de>.

RSBB - Reduced Space Branch-and-Bound Method

RSBB applies variable domain reductions and an underestimating module. Two versions run under Unix; one of these algorithms is described in Chapters 1 and 2 of Grossmann (1996). (Language: both versions in C++, they use external Fortran procedures.) Contact: T. Epperly, t.epperly@ic.ac.uk, http://www.ps.ic.ac.uk/~epperly/index.html.

SA - Simulated Annealing

In addition to the algorithm itself, this includes on-line interactive demonstration, and additional information on C++ classes , random number generation, Monte Carlo methods and Forth. A Nelder-Mead simplex method implementation is also available. (Languages: C, C++, Ada and Forth.) Contact: E. Carter, Taygeta Scientific Inc. <skip@taygeta.com>, http://www.taygeta.com/annealing/simanneal.html.

SAT - Global Optimization for Satisfiability Problems

Boolean satisfiability (SAT) problems can be directly transformed into unconstrained GOPs. These, in turn, can be solved by specifically tailored solvers. Workstation implementations. Contact: J. Gu <gu@enel.ucalgary.ca>.

SIGMA - Stochastic Integration Global Minimization Algorithm

The software package SIGMA is Algorithm 667; appeared in ACM-TOMS 14 (1988) 366-380. Includes also the code of several test GOPs. (Language: Fortran.) Contact: F. Aluffi-Pentini, V. Parisi and F. Zirilli <apzrm@itcaspur.caspur.it>, http://www.netlib.no/netlib/toms/667.

SIMANN - Simulated Annealing

This program implements the continuous simulated annealing global optimization algorithm described in Corana et al., ACM TOMS 13 (1987) No. 3, 262-280. Algorithm modifications and many details on its use can be found in Goffe et al., J. of Econometrics 60 (1994) No. 1-2, 65-100. (Language: Fortran.) Contact: B. Goffe, <bgoffe@whale.st.usm.edu>, http://www.netlib.no/netlib/opt/simann.f.

SOLVEX - Solver for Nonlinear Optimization Problems

For solving constrained and unconstrained nonlinear, multiobjective and GOPs. SOLVEX algorithm libraries include the methods listed below:
- unconstrained minimization: Hooke-Jeeves direct search, conjugate gradient method, Shor R-algorithm, Powell-Brent method
- general nonlinear programming problem: penalty functions, Lagrange function method, parameterization method
- global optimization: sequential set covering technique, simulated annealing, clustering algorithm
- multicriteria optimization: convolution methods, including goal programming, direct Pareto approximation.

Interactive use enhanced by a built-in problem editor and graphics capabilities. Contact: M.A. Potapov <potapov@sms.ccas.msk.su>.

TORUS - Stochastic Algorithm for Global Minimization with Constraints

It is a Monte Carlo algorithm, combined with annealing search principles. The software package TORUS is Algorithm 774; appeared in ACM-TOMS 21 (1995) 194-213. Includes also the code of several test GOPs. Contact: F. M. Rabinowitz, http://www.netlib.no/netlib/toms/744.

TRUST - Terminal Repeller Unconstrained Subenergy Tunneling

This method formulates the GOP as the solution of a deterministic dynamical system incorporating terminal repellers and a subenergy tunneling function. Benchmark tests comparing this method to other global optimization procedures are presented, with favourable results. The TRUST formulation leads to a simple stopping criterion. In addition, the structure of the equations enables an implementation of the algorithm in analog VLSI hardware (in the spirit of artificial neural networks) for further speed enhancements. Contact: B.C. Cetin, J. Barhen and W. Burdick; TRUST is described in JOTA 77 (1993) No. 1.

TVC - Toolbox for Verified Computing

Can be applied to the rigorous solution of nonlinear systems of equations, and to general unconstrained and bound-constrained GOPs. TVC is based on interval B&B and interval Newton methods, it also has automatic differentiation capabilities. The toolbox can be used on PCs, workstations and parallel computers. (Languages: PASCAL-XSC, C++.) Test problems have been solved up to one hundred variables. A driver program and hundreds of test examples are available from the author. Contact: D. Ratz <toolbox-xsc@math.uni-karlsruhe.de>, http://www.uni-karlsruhe.de/~iam, http://ourworld.compuserve.com/homepages/numerik_software.

UFO - Universal Functional Optimization

Interactive modular system for solving problems, and for algorithm development. Several types of GO methods - random search, continuation, clustering, and random plus local search - can be applied. For PCs. (Language: Fortran.) Contact: L. Luksan <luksan@uivt.cas.cz>, M. Tuma, M. Siska, J. Vlcek and N. Ramesova, ftp: uivt.cas.cz/pub/msdos/ufo.

UNICALC - Interval Branch and Bound Algorithm

UNICALC serves for bound-constrained GO; accepts also inequality and/or equality constraints and decision variables. Contact: A. Semenov, information available at ftp://ftp.iis.nsk.su/pub/ai/unicalc.

VerGO - Verified Global Optimization

VerGO is designed for rigorous bound (and approximate general constrained) GO of a twice continuously differentiable objective function. VerGO features include interval arithmetic, automatic differentiation, non-convexity test, monotonicity test, and local optimization. Tested on problems up to over 30 variables. DOS, OS/2, Linux and workstation versions. (Language: C++.) Contact: R. van Iwaarden <rvaniwaa@cs.hope.edu>, http://www.cs.hope.edu/~rvaniwaa/VerGO/VerGO.html.

VTT - Interval Arithmetic Research

The goals of the Interval Arithmetic, Constraint Satisfaction and Probability Project are summarized as follows:
- development of portable C++ libraries for interval programming tasks
- integration of the libraries to Microsoft Excel
- application in financial planning software products

(Platforms: C++, Excel.) Contact: S. De Pascale <Stefano.Depascale@vtt.fi>, http://www.vtt.fi/tte/.

4. Acknowledgements

The software review presented here is based to a significant extent on information kindly provided by colleagues working on GO and/or closely related areas. I would like to especially thank Arnold Neumaier and Simon Streltsov for the information collected on their WWW Global Optimization Pages (respectively, http://solon.cma.univie.ac.at/~neum/glopt/, and http://cad.bu.edu/go/). I also wish to thank Faiz Al-Khayyal for his valuable comments on the manuscript.

The space (and time) limitations of this review certainly have made it illusory to include 'all' existing software on this rapidly changing area: omissions are entirely possible, but absolutely unintentional... It is planned to continue this work, and to provide a more comprehensive and informative picture of the state-of-art, for the mathematical programming community. Comments and suggestions are most welcome: they will contribute to an 'unabridged' GO software review in the near future.

References

To avoid a superfluously long listing, the reference list is reduced to the most topical journal, and to several GO monographs and handbooks published in the past ten years.

Bomze, I.M., Csendes, T., Horst, R., and Pardalos, P.M., eds. (1996) Developments in Global Optimization. Kluwer Academic Publishers, Dordrecht / Boston / London.

Floudas, C.A. and Pardalos, P.M. (1990) A Collection of Test Problems for Constrained Global Optimization Algorithms. Lecture Notes in Computer Science 455, Springer, Berlin / Heidelberg / New York.

Floudas, C.A. and Pardalos, P.M., eds. (1992) Recent Advances in Global Optimization. Princeton University Press, Princeton.

Forg\'{o}, F. (1988) Nonconvex Programming. Akad/'{e}miai Kiad\'{o}, Budapest.

Grossmann, I.E., ed. (1996) Global Optimization in Engineering Design. Kluwer Academic Publishers, Dordrecht / Boston / London.

Hansen, E.R. (1992) Global Optimization Using Interval Analysis. Marcel Dekker, New York.

Horst, R. and Tuy, H. (1996) Global Optimization - Deterministic Approaches. Springer, Berlin / Heidelberg / New York. (3rd Edn.)

Horst, R. and Pardalos, P.M., eds. (1995) Handbook of Global Optimization. Kluwer Academic Publishers, Dordrecht / Boston / London.

Journal of Global Optimization (published since 1991, by Kluwer Academic Publishers).

Kearfott, R.B. (1996) Rigorous Global Search: Continuous Problems. Kluwer Academic Publishers, Dordrecht / Boston / London.

Michalewicz, Z. (1996) Genetic Algorithms + Data Structures = Evolution Programs. Springer, Berlin / Heidelberg / New York. (3rd Edn.)

Mockus, J. (1989) Bayesian Approach to Global Optimization. Kluwer Academic Publishers, Dordrecht / Boston / London.

Neumaier, A. (1990) Interval Methods for Systems of Equations. Cambridge University Press, Cambridge.

Pardalos, P.M. and Rosen, J.B. (1987) Constrained Global Optimization: Algorithms and Applications. Lecture Notes in Computer Science 268, Springer, Berlin / Heidelberg / New York.

Pint\'{e}r, J.D. (1996) Global Optimization in Action. Kluwer Academic Publishers, Dordrecht / Boston / London.

Ratschek, H. and Rokne, J.G. (1988) New Computer Methods for Global Optimization. Ellis Horwood, Chichester.

T\"{o}rn, A.A. and \v{Z}ilinskas, A. (1989) Global Optimization. Lecture Notes in Computer Science 350, Springer, Berlin / Heidelberg / New York.

van Laarhoven, P.J.M. and Aarts, E.H.L. (1987) Simulated Annealing: Theory and Applications. Kluwer Academic Publishers, Dordrecht / Boston / London.

Zhigljavsky, A.A. (1991) Theory of Global Random Search. Kluwer Academic Publishers, Dordrecht / Boston / London.