Stochastic Global Optimization: Problem Classes and Solution Techniques

Aimo Törn

Turku Centre for Computer Science and

Department of Computer Science, Abo Akademi University

Montaz Ali

Turku Centre for Computer Science and

Department of Computer Science, Abo Akademi University

Sami Viitanen

Department of Computer Science, Abo Akademi University

Turku Centre for Computer Science
TUCS Technical Report No 37
August 1996

ISBN 951-650-814-6
ISSN 1239-1891

Abstract

There is a lack of a representative set of test problems for comparing global optimization methods. To remedy this a classification of essentially unconstrained global optimization problems into unimodal, easy, moderately difficult, and difficult problems is proposed. The problem features giving this classification are the chance to miss the basin of the global minimum, the dispersion of minima, and the number of minima. The classification of some often used test problems are given and it is recognized that most of them are easy and some even unimodal. Working global optimization solution techniques treated are global, local, and adaptive search and their use for tackling different classes of problems is discussed. The problem of fair comparison of methods is then adressed. Further possible components of a general global optimization tool based on the problem classes and working solution techniques is presented.

Keywords: Global optimization, problem features, problem classes, test problems, solution techniques

1 Introduction

In this paper we discuss essentially unconstrained global optimization problems, i.e., find $\hat{f}^* = f(\hat{x^*})$, where $\hat{x^*} \in A \subset R^n$ so that $|\hat{f}^* - f^*| \leq \epsilon$, where f^* is the global minimum obtained in the interior of A. The region A is either a box or some other region easy to sample. We assume f to be given but not its analytical derivatives. We assume that the number of minimizers of f in A is finite. The methods we have in mind are those containing some probabilistic technique and the salient feature of these methods is that of the exploration of the search region A.

When presenting a new method authors illustrate the working of their method and compare its performance with that of some other algorithms on some test problems. In many cases the choice of test problems is quite random with the only systematic selection being over different values of n. Many of the test problems often used in the literature are trivially easy to solve and some of them are even unimodal and could thus be solved by applying a local optimization method from a single starting point.

Of course the choice of test problems should be systematic so that they represent different types of problems ranging from easy to difficult to solve. The failure in making such a choice may partly be caused by the lack of a suitable classification of problems according to some complexity measure and partly by the fact that the features of the test problems are not known.

This discussion shows that it would be important to be able to classify global optimization problems in order to test methods more systematically. This can then lead to a characterization of algorithms which is important in order to choose a suitable method given a problem with known features. Also such a classification could be the base for constructing an optimization tool that could characterize the problem at hand and then choose a suitable set of methods to apply.

We also address the problem of comparing methods.

2 Problem features and solution techniques

We are here discussing global optimization problem features and their contribution to problem complexity. We also recognize different techniques that are used in global optimization methods. The result of a crude problem complexity classification and working techniques for each class is presented in Table 1.

2.1 Problem Features

When solving (finding the global minimum of) a global optimization problem the outcome is dependent on the complexity of the problem. We postulate that the complexity is dependent on the following features of the problem: relative size p^* of the basin of f^* ($p^* = \mu(\text{basin of } f^*)/\mu(A)$), affordable number of global points N_g, the dispersion of the minima – clustered or scattered, and the number of local minima. The value of the expression $(1 - p^*)^{N_g}$ is the chance that basin of f^* is missed.

If the basin of f^* is large then the basin is easy to detect and such a problem is of course easier to solve than a problem with smaller such basin. As a rule it seems that the basin of the global minimum is the largest basin of all. This is for instance the case for the "standard" test problems Branin-Hartman6 (see Table 2). The explanation could be that if the minima basically have the same shape (same Lipschitz constant) then the size of the basins are larger for better minima.

By clustered we mean that the minima are arranged in such a way that the minimizer of a minimum is near minimizers of better minima, so that exploration near a minimum leads to detecting a better minimum and so on. Clustered minima therefore means that the basin of f^* may be found by such exploration even if the size of the basin is very small.

The number of minima and the size of the basin of f^* are normally not independent of each other. One would expect that the size is a decreasing function of the number of minima. However, it is an important feature of its own because local search will become increasingly ineffective for increasing number of local minima.

Of course there are other features which have an influence like expensiveness in evaluating $f(\cdot)$, unique or several global minimizers, the dimensionality n, and size and shape of A. If the function is very expensive to evaluate the number of affordable function evaluations is small. The same is also true for large n which generally will make an algorithm slower and thus influences on the affordable number of function evaluations. However, in both cases the increase in $(1 - p^*)^{N_g}$ will cover this increase in complexity. The size of A influences the complexity in the following way. Let A be the box $[0,1]^n$. If A is enlarged to $[0,2]^n$ then the volume of A grows with a factor 2^n. Sampling in the enlarged box will then increase $1 - p^*$ to maximally $1 - p^*/2^n$ so that this again is reflected in the chance to miss. A problem with several global minimizers is of course harder if all minimizers are to be found and may also affect efficiency because of convergence problems for some methods.

Problem features					Working techniques		
Class	complexity	$(1-p^*)^{N_g}$	disp	#mins	glob	local	adapt
U	unimodal	0	0	1	(+)	++ld	+
E_1	easy	$<$	$<>$	$<$	+	+ld	+
E_2		$<$	$<>$	$>$	+	+gd	+
C_1	moderate	$>$	$<$	$<$	+	++ld	+
C_2		$>$	$<$	$>$	+	++gd	+
D_1	difficult	$>$	$>$	$<$	++	+ld	0
D_2		$>$	$>$	$>$	++	+gd	0

Table 1: GO problem classes and working solution techniques

2.2 Solution techniques

Solution techniques used in global optimization are: *global* technique, *local* technique, and *adaptive* technique.

The global technique is responsible for exploring the whole area of interest A and in this way secure that a point in the basin of the global minimum is found.

The local technique is used to find better points in the vicinity of some (good) point in order to improve the accuracy of a solution. In the case of many clustered minima the local technique could also be responsible for finding better minima near promising minima.

The local technique is normally responsible for exploring promising parts of A. This can also be done in another way. By an adaptive technique we mean that the global technique is gradually sampling more points in the regions where good points have already been found. This technique is used in methods where no local technique is used so that the global technique gradually turns into a local technique. We think that using an explicit local technique is more rewarding than such a technique. Another motive for using adaption is based on the assumption that the minima are clustered. If this is the case then it is rewarding to focus the search on neighborhoods of promising points.

Other techniques used are single working point technique, multistart, clustering, converging set. These techniques are not generally applied but rather relate to some special methods.

3 Problem Classes and Working Techniques

In Table 1 we present the relation between problems and working techniques. There are six classes presented in increasing order of complexity. The problem features have been presented in such a way that a small value ($<$) means small complexity. For the minima, $<$ means few minima; for $(1-p^*)^{N_g}$, $<$ means small chance of missing; and for dispersion, $<$ means clustered, $>$ scattered.

In the technique part of the table a $+$ means that the technique is used, $++$ means that the main effort is in using this technique. In the column for the local technique the notation *ld* means local descent and *gd* means global descent, i.e., local improvement for instance by sampling so that the local technique may escape inferior minima.

The adaptive technique is generally applicable to problems where either the chance of missing the basin of f^* is small or the minima are clustered. We note where adaption is favorable by a $+$ in the column named 'adapt'. For difficult problems where the minima are scattered applying an adaptive technique will in general lead to an increase in the chance to miss the global minimum and therefore adaption should not then be applied.

Table 2 shows some examples of problem members of each class. The characterization of sample problems is partly based on experiences made in solving global optimization problems with Control Random Search Techniques, [ATV96].

The values in the column p^* in Table 2 have been obtained by applying a local descent algorithm to 1000 points randomly distributed over A.

3.1 Unimodal Problems

Unimodal problems are not global optimization problems and therefore actually no global technique needs to be applied. The (+) in the column for the global technique indicates the possibility to use several starting points in order to guarantee that the local optimization method really converges to the global minimum. Global optimization test problems belonging to this class are Powell, which analytically can be proved to have only one minimizer, and Kowalik, which in all our experiments were found to have a single minimizer. The use of local optimization problems as test problems in global optimization is hard to motivate. Possibly the unimodal feature of these problems were not known when they were used.

3.2 Easy Problems

The easy problems are characterized by small chance to miss the basin of the global minimum. This means that either the basin is large or that enough points to be sampled in order for the chance to miss the basin is small. For large basins the strategy could be to sample a small number of global points uniformly in A and then start local optimizations from some promising points. Most global optimization methods should work. For few minima methods designed to find all local minima (e.g. clustering techniques) could be used. There are many test problems belonging to this class, the most known are the "standard test problems" Branin, Goldstine-Price, Shekel 5,7,10 and Hartman 3,6. The problems Hosaki and Levy and Shubert3 also belongs to this class.

For small basins more effort have to be devoted to the global part. An example of problems belonging to this class is the problem Shubert5 with 15^5 minima and $p^* = 0.05$. For 90 global points the probability to miss the global basin is less than 0.01.

For many minima global descent rather than local decent should be used.

3.3 Moderately Difficult Problems

The moderately difficult problems are characterized by that the probability to miss is large but that the minima are clustered. This means that the vicinity of a promising point (a point with relative small function value) in a basin of a minimum has other promising points in basins of other minima. This means that an adaptive technique works well. The CRS method CRS(q,β) [ATV96] could be recommended. The Griewank problems are belonging to this class. They have many local minima with the global minimum in origin and many other nearby. There seems to be a lack of problems in the literature belonging to the class C_1.

3.4 Difficult Problems

The difficult problems are characterized by large chance to miss and scattered minima. This means that the detection of a point in the basin of the global minimum must totally rely on sampling in A. There is no reward in using an adative technique for the global part because the minima are scattered. The number of minima could either be small or large and this will affect which local technique is to be used. For a small number of minima it will be effective to use a local descent method when a promising point is found by

Class	Function	n	Region A	#min	p^*	f^*	Ref
U	Kowalik	4	$0 \leq x_i \leq 0.42$	1	1.00	0.00	JK95
	Powell	4	$-10 \leq x_i \leq 10$	1	1.00	0.00	JK95
E_1	Branin	2	$-5 \leq x_1 \leq 10$	3	1.00	0.40	TZ89
			$0 \leq x_2 \leq 15$				
	Goldprice	2	$-5 \leq x_i \leq 10$	4	0.40	3.00	TZ89
	Shekel5	4	$0 \leq x_i \leq 10$	4	0.35	-10.15	TZ89
	Shekel7	4	$0 \leq x_i \leq 10$	7	0.35	-10.40	TZ89
	Shekel10	4	$0 \leq x_i \leq 10$	10	0.35	-10.54	TZ89
	Hartman3	3	$0 \leq x_i \leq 1$	4	0.70	-3.86	TZ89
	Hartman6	6	$0 \leq x_i \leq 1$	4	0.70	-3.32	TZ89
	Hosaki	2	$0 \leq x_1 \leq 5$	2	0.65	-2.35	BU74
			$0 \leq x_2 \leq 6$				
E_2	Levy10	10	$-10 \leq x_i \leq 10$	10^{10}	0.85	0.00	JK95
	Schubert3	3	$-10 \leq x_i \leq 10$	$\approx 5^3$	0.35	0.00	DA91
	Schubert5	5	$-5 \leq x_i \leq 5$	$\approx 15^5$	0.05	0.00	DA91
	Many-	3	$0 \leq x_{1,2} \leq 1.3$				AST96
	body	6	$0 \leq x_3 \leq \pi$				
			$i > 3 \mid x_i \mid \leq 1.5$				
C_1							
C_2	Griewank	2	$-100 < x_i < 100$	≈ 500		0.00	TZ89
	Griewank	10	$-600 < x_i < 600$	$O(10^3)$		0.00	TZ89
D_1							
D_2	M-body	≥ 9	(see above)				AST96

Table 2: GO problem classes

the global technique. Here clustering techniques or other techniques which try to find all local minima could be applied.

Many of these problems are probably unsolvable with affordable effort and one can only hope to find a good local solution. Examples of difficult problems are Many-body Problems [AST96]. These become increasingly difficult with increasing n because of the increasing number of minima and growing $(1 - p^*)$. Some of these $n = 3, 6$ are solvable by using some thousands of function evaluations, and could be characterized as easy, other require several more times of function evaluations for a serious solution effort and can be characterized as difficult. There seems to be a lack of problems in the literature belonging to the class D_1.

4 Comparing Methods

For comparing methods we propose that test problems should represent different classes of problems, for instance Branin, Shekel5, Shubert5, Griewank, and some Many-body problems or other problems from class D.

For a method the outcome of a solution effort is dependent on the stopping condition used because this will determine how much work is put down to find the solution. We therefore touch on this problem below.

Because the stopping conditions normally are not compatible the comparison of methods must be empirical. How this should be properly done is discussed.

4.1 Stopping Conditions

Every method must use some stopping condition. Stopping conditions are sometimes based on theoretical convergence properties. Because a global optimization problem generally cannot be solved for sure the convergence properties are at best probabilistic which means that some method will find the global minimum with a probability that approaches 1 as the algorithm runs on. The convergence is dependent on that a point in the basin of the global minimum is found and that a local algorithm can successfully find the minimum when started from such a point. Even if a method converges in probability it is not normally possible to estimate the probability that the global minimum is found when the algorithm stops and N function evaluations have been made. An exception is multistart if we assume that the local technique always is successful and we know p^* or a lower bound of it [TZ89]. It is trivial to make any method converge in probability by adding some random sampling element that is always applied but possibly with a probability decreasing over time in order not to degrade the efficiency of the method. This means that the question whether a method has theoretical convergence properties or not is not essential from a practical point of view.

4.2 Empirical Comparison

Methods are normally compared on their efforts needed to find the global minimum (eg. number of function evaluations N or cpu time). A method is considered better than another method if it solves the problem with fewer function evaluations or in less cpu time.

When making empirical comparisons between methods it is normally not recognized that a probabilistic global optimization method M applied to a problem P is a mapping $(M, P) \rightarrow (E, q)$, where E is the effort applied in the solution process and $q \in [0, 1]$ is the probability that the global minimum is found. The effort could either be measured as cpu time and/or as number of function evaluations N. This means that when comparing two methods M_1 and M_2 for a problem P the pairs (E_1, q_1) and (E_2, q_2) are to be compared. If then either one pair dominates the other or $q_1 = q_2$ or $E_1 = E_2$ such a comparison is possible otherwise not. The pair (E_1, q_1) dominates (E_2, q_2) if they are not equal and $E_1 \leq E_2$ and $q_1 \geq q_2$. In order to compare two methods the easiest way is thus to fix E and apply the methods repeatedly in solving some problem and then record averages for q_1 and q_2 and compare them. This should then be done for a range of E. Alternatively one could fix q and try to find such parameters for the methods so that q is achieved on the average. Both of these approaches may be difficult to realize because the different stopping conditions used by the algorithms are not easily related to E or q.

In conclusion we think that comparisons of methods reported in the literature are normally not fair in the way explained above. Some are rather heuristic based on incompatible stopping conditions giving an outcome which as a rule seems to be in favor for the new method presented.

5 Elements of a Generally Applicable Global Optimization Tool

Given a global optimization problem the features of the problem are not always known. What strategy should then be applied to solve the problem? We think that a characterization of the problem is part of the solution and therefore the solution strategy should be to apply methods that would reveal the features. We next address exploring the features of a given problem, see Table 1.

The method to use for finding #min is multistart, i.e., starting a local minimization algorithm from m randomly sampled starting points in A. If most of the minimizations arrive at different solutions many minima can be expected, otherwise few. Other information obtained is the variation of the function over A and the effort needed (function evaluations) for local minimization. Further the affordable number of function evaluations in solving the problem can be estimated. By analysing the minima obtained using some clustering technique some indication of their dispersiveness can be obtained.

The feature p^* can of course not be estimated because this means that we know for sure we have solved the problem which might not normally be possible to assure. One possible way to proceed is to make some assumption about the lower limit of p^*. Such an assumption could be based on practical information, i.e., the usefullness of a solution might be dependent on that the solution is stable so that a variation inside the needed accuracy in the decision variables x means small variations in the function value.

The requirements on methods to include for solving global optimization problems with unknown features is that it should allow the user to switch the mode of exploration according to the initially known problem features and those subsequentially found during runs, e.g. switch between local and global, and utilize adaptive techniques if deemd favorable. Possible candidates to include in such a pool would be CRS($\Delta, N-M$) (CRS3 [P87]) and CRS(q, β) [AST96]. These methods, as all CRS methods, also have a very natural and easy to understand stopping condition. Another candidate would be Iterative TGO [TV96] partly because of the extra information about probable local minima obtained during a run.

Based on these toughts it should be possible to construct a global optimization tool that would support in obtaining a solution by in addition to offering several global optimization methods also would include an overall strategy in solving global optimization problems.

References

[AST96] M. Ali, C. Storey and A. Törn, Application of some Recent Stochastic Global Optimization Algorithms to Practical Problems, submitted to *Journal of Optimization Theory and Applications*, 15 pp.

[ATV96] M. Ali, A. Törn and S. Viitanen, Controlled Random Search Algorithms for Unconstrained Global Optimization, submitted to *Journal of Global Optimization*, 8 pp.

[BU74] G.A. Bekey and M.T. Ung, A Comparative Evaluation of Two Global Search Algorithms, *IEEE Trans. Syst. Man, Cybern.*, SMC-4, No. 1, 112-116.

[C84] V. Černy, Minimization of Continuous Functions by Simulated Annealing, Research institute for Theoretical Physics, University of Helsinki, Preprint No. HU-TFT-84-51.

[DA91] A. Dekker and E. Aarts, Global Optimization and Simulated Annealing, *Mathematical Programming* 50, 367-393.

[JK95] C. Jansson and O. Knüppel, A Branch and Bound Algorithm for Bound Constrained Optimization, *Journal of Global Optimization*, 7, 297-331.

[P87] W.L. Price, Global Optimization Algorithm for a CAD Workstation, JOTA 55, 133-146.

[TV96] A. Törn and S. Viitanen, Iterative topographical global optimization, In: C.A. Floudas and P.M. Pardalos (Eds.), State of the Art in Global Optimization, Princeton University Press, 353-363.

[TZ89] A. Törn and A. Žilinskas, Global Optimization, Lecture Notes in Computer Science 350, Springer-Verlag, 255 pp.

Simulated

Turku Centre for Computer Science
Lemminkäisenkatu 14
FIN-20520 Turku
Finland

http://www.tucs.abo.fi

University of Turku
- **Department of Mathematical Sciences**

Åbo Akademi University
- **Department of Computer Science**
- **Institute for Advanced Management Systems Research**

Turku School of Economics and Business Administration
- **Institute of Information Systems Science**