INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2002; **53**:2239–2252 (DOI: 10.1002/nme.400)

Studies of multi-start clustering for global optimization

W. Tu[1,*,†] and R. W. Mayne[2,‡]

[1]*Educational Communications, SUNY Upstate Medical University, 766 Irving Avenue, Syracuse, NY 13210, U.S.A.*
[2]*Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, Buffalo, NY 14260, U.S.A.*

SUMMARY

Global/multi-modal optimization problems arise in many engineering applications. Owing to the existence of multiple minima, it is a challenge to solve the multi-modal optimization problem and to identify the global minimum especially if efficiency is a concern. In this paper, variants of the multi-start with clustering strategy are developed and studied for identifying multiple local minima in nonlinear global optimization problems. The study considers the sampling procedure, the use of Hessian information in forming clusters, the technique for cluster analysis and the local search procedure. Variations of multi-start with clustering are applied to 15 multi-modal problems. A comparative study focuses on the overall search effectiveness in terms of the number of local searches performed, local minima found and required function evaluations. The performance of these multi-start clustering algorithms ranges from very efficient to very robust. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: global; optimization; multi-modal; Hessian; multi-start; clustering

1. INTRODUCTION

In this paper, the global optimization problem is considered subject to variable bound constraints (the box-constrained problem):

$$\min_{\mathbf{x}\in D} f(\mathbf{x}), \quad D \subset R^n \tag{1}$$

where $f(\mathbf{x})$ is a multi-modal objective function and D is the feasible domain, defined by variable bounds. A point $\mathbf{x}^*$ is a global minimum if $f(\mathbf{x}) \geqslant f(\mathbf{x}^*)\ \forall \mathbf{x} \in D$, or a local minimum if $f(\mathbf{x}) \geqslant f(\mathbf{x}^*)\ \forall \mathbf{x} \in D \cap X$, where X is the neighbourhood of $\mathbf{x}^*$. If bounds are not tight

*Correspondence to: Weizhen Tu, Educational Communications, SUNY Upstate Medical University, 766 Irving Avenue, Syracuse, NY 13210, U.S.A.
†E-mail: tuw@upstate.edu
‡E-mail: mayne@eng.buffalo.edu

Contract/grant sponsor: State of New York/UUP Affirmative Action Committee
Contract/grant sponsor: Computing and Information Technology of the State University of New York at Buffalo

Received 5 February 2001
Revised 20 April 2001

at the minimum, the problem can be considered as unconstrained. Traditionally, nonlinear programming (NLP) methods have been developed to aim at a local minimum. Although a global minimum must also be a local minimum, there is no mathematical criterion for deciding whether a particular local minimum is indeed the global minimum. There may exist several local minima and the corresponding function values may differ substantially. It has been proven that a general global optimization problem cannot be solved in a finite number of steps [1]. It is a considerable challenge to solve the multi-modal optimization problem and identify the global minimum from both mathematical and computational viewpoints especially if efficiency is a concern.

Global optimization algorithms can be broadly classified as deterministic or stochastic depending on whether they incorporate any stochastic elements to solve the problem [2, 3]. The multi-start approach [2] is one of the well-known stochastic methods and tries to find multiple local minima by starting local minimization procedures from a set of random starting points distributed uniformly over the feasible domain. Several of its variants have been reported in the literature such as multi-start with clustering [4–7], domain elimination [8], zooming [8], and repulsion [9]. One of the advantages of multi-start is that it has the potential to find all local minima. But, the basic multi-start method has the tendency to be inefficient because it inherently causes extra executions of the local search procedure and particular minima may be located several times. However, if sample points can be grouped into clusters that correspond to regions of attraction, where a region of attraction $R(\mathbf{x}^*)$ of a local minimum $\mathbf{x}^*$ is defined as a subset of the feasible domain within which a given local search procedure starting from any point converges to $\mathbf{x}^*$ [5], then ideally only one local search is required in each cluster. This is the basic idea of multi-start with clustering.

In this work, variants of the multi-start with clustering approach for solving unconstrained global optimization problems are studied, developed and implemented. They differ from existing algorithms in the way that sample points are manipulated and clusters are identified. Specifically, the Hessian matrix is considered as a guide to clustering decisions, simulated annealing is used in the sampling process and the 'ISO-OCT' technique is adapted from the pattern recognition literature for use in cluster analysis. In the sections which follow: Section 2 introduces the use of Hessian information in multi-start clustering; Section 3 discusses sampling methods; Section 4 discusses clustering algorithms; Section 5 presents a multi-start clustering strategy for unconstrained global optimization; Section 6 describes and summarizes numerical studies of the algorithm character; and Section 7 reports on comparative results with previously published algorithms.

2. USE OF HESSIAN INFORMATION IN MULTI-START CLUSTERING

Assume that the function to be minimized is smooth and twice differentiable. As known from the necessary and sufficient conditions for local optimality, at a local minimum the gradient of the objective function is zero and the Hessian matrix is positive definite. Intuitively, we can imagine that in the neighbourhood of local minima, the objective function is convex and any two isolated local minima must be separated by a region where the function is non-convex, that is, the Hessian matrix is either negative definite or indefinite. If we process the sample points by discarding those points with non-positive definite Hessian matrices, the resulting clusters should be well-defined and more easily and accurately identifiable

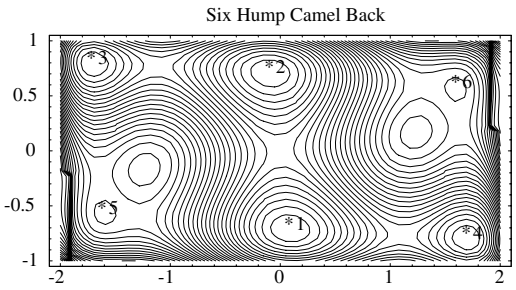

Figure 1. Contour plot of six-hump camel back.

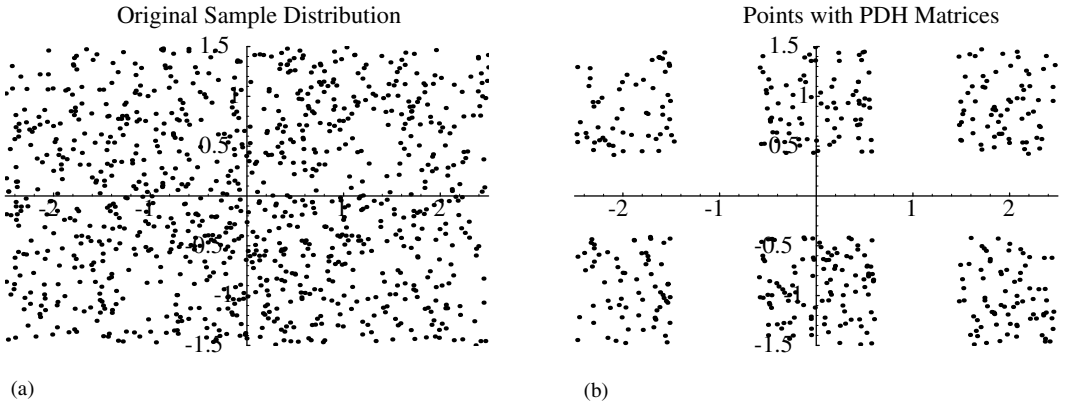

Figure 2. (a) Original sample distribution; and (b) Points with positive definite Hessian matrices.

(Figures 1 and 2, e.g.). The efficient extraction of Hessian information is a natural concern in this process.

The symmetric rank one (SR1) formula has been used to estimate the Hessian matrix, because the SR1 update neither requires quasi-Newton directions nor an exact line search to generate a reasonable Hessian approximation [10, 11]. Moreover, it has been demonstrated that the SR1 update tends to converge to the true Hessian in terms of definiteness and numerical value regardless of the 'definiteness' of the Hessian [10, 11]. The SR1 formula for the Hessian estimation is as follows [10]:

$$H^{k+1} = H^k + \frac{(Y^k - H^k S^k)(Y^k - H^k S^k)^{\mathrm{T}}}{(Y^k - H^k S^k)^{\mathrm{T}} S^k} \tag{2}$$

$$B^{k+1} = B^k - \frac{(B^k Y^k - S^k)(B^k Y^k - S^k)^{\mathrm{T}}}{(Y^k)^{\mathrm{T}}(B^k Y^k - S^k)} \tag{3}$$

where $Y^k = \nabla f(X^{k+1}) - \nabla f(X^k)$ and $S^k = X^{k+1} - X^k$. The approximation of the Hessian matrix and its inverse at the $(k+1)$th and kth iterations are given by H^{k+1}, H^k and B^{k+1}, B^k, respectively. For an n-dimensional quadratic function, it takes at most $n+1$ iterations to approximate

the Hessian matrix. For each sample point, a sample-point-based approach to estimating the Hessian uses its n closest points for $n+1$ iterations. Estimating the Hessian matrix for N sample points requires N gradient evaluations, N sort operations and $N \times (N-1)$ distance calculations. When function and gradient evaluations are expensive, the cost of computing distances and sorting is negligible.

Eigenvalues are used to check the definiteness of the Hessian matrix at each sample point. Geometrically speaking, the eigenvalues of an Hessian matrix correspond to the lengths of the axes of an ellipsoid. The largest eigenvalue corresponds to the shortest axis of the ellipsoid, while the smallest eigenvalue corresponds to the longest axis. To make an effective comparison of eigenvalues from point to point, scaled eigenvalues are used at each point. A scaled eigenvalue for one point is defined as the ratio of each original eigenvalue over the sum of all the eigenvalues for that point, so that,

$$\lambda_i^s = \frac{\lambda_i}{\sum_{i=1}^{n} \lambda_i} \tag{4}$$

Then the points with at least one negative eigenvalue or with relatively smaller scaled positive eigenvalues are considered to be 'not positive definite' and may be discarded from the clustering process of the continuing search.

3. SAMPLING METHODS

Two sampling methods are considered, a one-temperature simulated annealing (SA) algorithm and a simple random search. Simulated annealing is a stochastic global optimization method [12]. It iterates as a series of random search procedures performed at a decreasing sequence of the control parameters (temperatures). At each iteration, a new point $\mathbf{x}'$ is generated by randomly perturbing an existing point x. If $f(\mathbf{x}') < f(\mathbf{x})$ then the new point is accepted. Otherwise, the new point is accepted according to a probability that is a function of the temperature. The one-temperature SA algorithm uses one fixed temperature and iterates only once. It differs from a simple random search which accepts all points sampled.

Several methods for sample reduction and concentration have been reported in the literature, including: (1) eliminating sample points with larger function values [5]; (2) classifying sample points based on a gradient criterion [6]; and (3) one or a few step local searches [4]. One strategy considered here processes sample points by eliminating points with non-positive definite Hessian matrices.

The function value and gradient provide point-wise information, while the Hessian reflects the nature of a function over a region. The objective function is expected to be convex in the neighbourhood of a local minimum in unconstrained problems. Thus discarding points with a non-positive definite Hessian matrix offers a potential for robust and efficient behaviour in finding all local minima, because it theoretically matches each cluster to the region of attraction of each local minimum.

However, the region of attraction of a local minimum is very small for problems with a very rough surface. Then depending on the scale of the roughness, the use of Hessian information may not be effective, in which case, eliminating sample points with larger function values may be a preferred approach to reducing sample size and to preparing the sample for cluster analysis.

4. CLUSTER ANALYSIS

Generally speaking, cluster analysis is a mathematical method for generating classes without *a priori* knowledge of prototype classification. Cluster analysis is used in many applications such as pattern recognition and image processing. Assume that there are M clusters in space S. The process of clustering can be formally stated as seeking the regions $S_1, S_2, \ldots, S_m$, which satisfy the following conditions [13]:

$$S_i \cap S_j = \emptyset \quad \text{for } i \neq j \quad \text{and} \quad S_1 \cup S_2 \ldots \cup S_m = S \tag{5}$$

Cluster analysis algorithms classify objects into clusters by natural association according to similarity measures. Euclidean distance is the simplest and most frequently used measure and is represented by

$$D^2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{x}_i - \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \sum_{k=1}^{n} (x_{ik} - x_{jk})^2 \tag{6}$$

where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^{\mathrm{T}}$. It is expected that the intraset distance (within clusters) should be small, whereas the interset distance (between clusters) should be large. In this study, we use the iterative self-organizing optimal clustering technique (ISO-OCT) which is a variant of the iterative self-organizing data analysis technique (ISODATA) [13] and incorporates an optimal cluster seeking criterion function to optimize the clustering process and obtain the natural clusters of the sample points [14]. The algorithm assigns sample points to clusters according to their distance from the initially chosen cluster centres. Then it reorganizes clusters by splitting and merging clusters based on two criteria:

1. If the distance between two cluster centres is less than the minimum distance allowed (δ), then merge the two clusters into one;
2. If the largest standard deviation of a cluster (used as an intraset measure) is greater than the maximum value allowed (σ_s) and if the average distance of the samples in a cluster S_j from their corresponding cluster centre is greater than the overall average distance of the samples from their respective cluster centres, then split that cluster.

The initial cluster centres are chosen in the following way: (1) sort function values of the NP sample points in an ascendant order, that is, $f(\mathbf{x}_1) \leqslant f(\mathbf{x}_2) \leqslant \cdots \leqslant f(\mathbf{x}_{\mathrm{NP}})$; (2) choose three points as the initial cluster centres: $\mathbf{z}_1 = \mathbf{x}_1$, $\mathbf{z}_2 = \mathbf{x}_{\mathrm{NP}/2}$ and $\mathbf{z}_3 = \mathbf{x}_{\mathrm{NP}}$. The values for δ and σ_s are computed as follows:

$$\delta = 0.2 \left\{ \frac{1}{2} \left(\frac{1}{n} \sum_{i=1}^{n} |u_i - l_i| + \frac{1}{n} \sum_{i=1}^{n} |x_i^{\max} - x_i^{\min}| \right) \right\} \tag{7}$$

$$\sigma_s = 0.5\delta \tag{8}$$

where n is the dimension of the design space, u_i and l_i $(i = 1, 2, \ldots, n)$ are upper and lower variable bounds, and $x_i^{\max} = \max\{x_i^1, x_i^2, \ldots, x_i^{\mathrm{NP}}\}$ and $x_i^{\min} = \min\{x_i^1, x_i^2, \ldots, x_i^{\mathrm{NP}}\}$.

The optimal cluster seeking criterion function (a similarity measure) is defined as [14]:

$$R_{ij} = \frac{D_{ii} + D_{jj}}{D_{ij}} \tag{9}$$

where D_{ii} and D_{jj} are the dispersions for clusters i and j, respectively; and D_{ij} is the distance between clusters i and j. D_{ii} and D_{ij} can be computed as:

$$D_{ii} = \left[\frac{1}{N_i} \sum_{j=1}^{N_i} \|\mathbf{x}_j - \mathbf{z}_i\|^2 \right]^{1/2} \quad \text{and} \quad D_{ij} = \left[\sum_{k=1}^{n} (z_{ki} - z_{kj})^2 \right]^{1/2} \tag{10}$$

where z_{ki} is the kth component of cluster centre $\mathbf{z}_i$. For each cluster i, the similarity parameter for its most similar cluster is designated as $R_i = \max_{j=1}^{N_c} \{R_{ij}, j \neq i\}$, where N_c is the number of clusters. Equation (9) shows that the smaller the dispersion of a cluster or the larger the distances among clusters, the lesser is the similarity of clusters. The average measure of the most similar clusters $\bar{R}$, is then computed as:

$$\bar{R} = \frac{1}{N_c} \sum_{i=1}^{N_c} R_i \tag{11}$$

The cluster algorithm starts with an arbitrary number of clusters. Different values of $\bar{R}$ can be obtained for different cluster configurations in each iteration. The configuration of clusters corresponding to the smallest values of $\bar{R}$ is the most appropriate configuration [14].

Several cell-based clustering methods have been used in global optimization such as multi-level single linkage (MLSL) [5], multi-level mode analysis (MLMA) [5] and vector quantization (VQ) [6]. These methods identify clusters sequentially with the seed cells containing the local minima found by local searching, and use Bayesian stopping rules [5] to terminate the whole process of sampling, clustering and performing local searches. The difference among them is that VQ uses Voronoi cells as polytopes for clustering, while MLSL and MLMA are based on the spiral search technique which uses hypercubes. The ISO-OCT strategy identifies all clusters simultaneously before starting any local searches. It uses the ratio of the intercluster distance vs dispersion within a cluster as a criterion function to guide the splitting and merging of clusters to improve the cluster configuration.

5. THE MULTI-START WITH CLUSTERING STRATEGY FOR UNCONSTRAINED GLOBAL OPTIMIZATION

The multi-start with clustering strategy studied here begins by sampling the design domain via a one-temperature SA algorithm based on Hsueh's implementation [15] or alternatively with a simple random search procedure. Then the sample points can be reduced in one of two ways: (1) discarding points with non-positive definite Hessian matrices; (2) discarding points with larger function values. After this reduction, the cluster analysis technique is applied to the reduced sample points to identify clusters. Finally, a gradient-based quasi-Newton local search procedure starts in each cluster to find the local minimum and potentially the global minimum. The strategy can be summarized in the following framework:

Step 0. Input the dimension n, sample size N, variable lower/upper bounds l_j and u_j ($j = 1, \ldots, n$) for the sample procedure. Specify the maximum number of clusters M, minimum number of samples in a cluster η, maximum number of pairs of cluster centres that can be merged L, and the maximum number of iterations I for the ISO-OCT procedure. If the one-temperature SA is used, input the temperature T_0.

Step 1. Generate N random points $\mathbf{x}_i$ using the one-temperature SA algorithm or a simple random search procedure, store N_{sel} accepted sample points (for random search $N_{\text{sel}} = N$) and their function values in arrays X_{sel} and F_{sel}.
Step 2. Compute the clustering parameters δ and σ_s using Equation (7) and (8).
Step 3. If the flag for using the Hessian is false, then discard 20 per cent of the sample points with larger function values and go to Step 7. Otherwise, go to Step 4.
Step 4. Compute the gradient $\mathrm{d}g(\mathbf{x}_i)$ $(i = 1, \ldots, N_{\text{sel}})$ using either a user supplied gradient routine or the finite difference method.
Step 5. Find n closest points of $\mathbf{x}_i$ and compute the Hessian matrix $H(\mathbf{x}_i)$ $(i = 1, \ldots, N_{\text{sel}})$ using Equations (2) and (3).
Step 6. Compute the eigenvalues (using the subroutine EVLSF of the IMSL libraries [16]). Then compute the scaled eigenvalues λ_{ij}, $j = 1, \ldots, n$, $i = 1, \ldots, N_{\text{sel}}$ using Equation (4). If any $\lambda_{ij} < 0$, discard the point i, else store the largest λ_{ij} $(j = 1, \ldots, n)$ in an array λ_{pd}. Sort sample points by λ_{pd} in an ascending order and retain the 80 per cent of the points with the largest scaled eigenvalues.
Step 7. Identify clusters by applying the clustering analysis procedure ISO-OCT to the reduced sample points.
Step 8. Perform a local search procedure (BCONG or BCONF of IMSL [16]) starting from the point with the smallest function value in each cluster.
Step 9. Stop. The points with the smallest function values are candidate global minima.

The above strategy including Steps 4–6 has been implemented in FORTRAN 77 as an 'H-based' (Hessian-based) program. If the Steps 4–6 are skipped, the program is referred to as 'DC' for direct clustering. No special stopping conditions are required, as the program stops when all local searches terminate. The BCONF/BCONG routine of IMSL minimizes a function of n variables subject to bounds on the variables using the BFGS quasi-Newton method [16]. To save function evaluations in the finite difference gradient calculation, the function value of each sample point accepted by the sample procedure is stored and passed into the gradient computation subroutine. Thus, for a sample size of N points, only $N \times n$ not $(N+1) \times n$ extra function evaluations are incurred for the finite difference gradient calculations.

6. NUMERICAL RESULTS

Variations of the multi-start strategy have been tested on 15 problems selected from the global optimization literature. These problems represent a mixture of reasonably behaved functions with a few minima and highly non-linear functions with many minima. Problems 13–15 are extensions of the Shekel family [3] to ten dimensions. A summary of the test problems is listed in Table I, where n is the dimension of the problem, NL/NGM is the number of local/global minima, and $F_{\text{G}}/F_{\text{L}}$ is the smallest/largest function value among all of the minima. Full descriptions of the test problems can be found in Reference [18].

The multi-start variations considered include different sampling methods (one-temperature SA vs random search), and clustering strategy (using Hessian information before clustering or direct clustering). All tests were performed on a 250MHz Sun Ultra Enterprise 3000 computer.

Complete numerical results were reported in Reference [18], including the number of function evaluations in the sample process and local searches; and the number of gradient

Table I. Summary of 15 multi-modal problems.

Problem	n	NL	NGM	F_G	F_L	Referred to as
1	2	4	1	3.0	840	Goldstein–Price (GP) [3]
2	2	3	1	0.0	0.298640	Three-Hump Camel back (CB3) [2]
3	2	6	2	−1.03163	2.104	Six-Hump Camel back (CB6) [8]
4	2	3	3	0.397887	0.397887	Branin (BR) [3]
5	2	4	4	0.0	0.0	Himmelblau (HM) [17]
6	3	3	3	−3.86278	−1.00082	Hartman-3 (H3) [3]
7	6	2	2	−3.32237	−3.20461	Hartman-6 (H6) [3]
8	4	5	1	−10.15319	−2.63047	Shekel-5 (S5) [3]
9	4	7	1	−10.40294	−1.83759	Shekel-7 (S7) [3]
10	4	10	1	−10.5364	−1.67655	Shekel-10 (S10) [3]
11	4	Unknown	1	0.0003075	0.002119	Kowalik (KL) [8]
12	6	Unknown	2	−1.00	−0.85219	Evtushenko (EV) [8]
13	10	5	1	−10.15320	−2.63047	Shekel-ten-5 (ST5) [18]
14	10	7	1	−10.40294	−1.83759	Shekel-ten-7 (ST7) [18]
15	10	10	1	−10.53641	−1.67655	Shekel-ten-10 (ST10) [18]

calculations in Hessian matrix estimation and local searches. A summary of the results is given in Table II where analytical gradient evaluations are counted as $N \times n$ function evaluations. The sample size for each method is included in parentheses. All results are the average of five independent runs. However, the rows labelled lnf/tnf report the average and total number (average/total) of local minima found in the five runs.

With the one-temperature SA, a sample size of 400 points was chosen as a reasonable compromise between computational effort and the number of accepted sample points after preliminary experiments were run with 1000, 400 and 100 samples. In columns 4 and 5 of Table II, for problems with a few local minima and relatively large regions of attraction such as Goldstein–Price (GP), Camel-3 (CB3) and Camel-6 (CB6), Himmelblau (HM), and Hartman-3 (H3) and Hartman-6 (H6), the H-based approach is more effective at identifying the regions of attraction of local minima. It found more local minima per run than direct clustering (DC). The number of local searches it performed also represents the number of local minima more closely than the DC approach. Although the H-based approach is associated with increased function evaluations, it may be useful in cases where local minima are of particular interest as perhaps in mechanical design applications. In cases where there are many minima (rough surfaces, e.g. the Kowalik, KL, and Evtushenko, EV, functions designed for global minimization testing) or where the region of attraction is very small (perhaps with deep basins, e.g. the Shekel family of functions), it is difficult to find points with positive definite Hessian matrices at a practical sampling scale. This makes the H-based method less effective than the direct clustering method in locating multiple local minima. Both the H-based and DC methods are able to locate the global minima for all problems. Overall, the H-based method incurs more function evaluations than direct clustering because of the required function evaluations for Hessian matrix estimation. This becomes more evident when the number of design variables increases.

Columns 4 and 5 of Table II also show that the one-temperature SA procedure consumes a significant fraction of function evaluations. Based on this observation, an experiment with a simple random search procedure has been considered to reduce the sample size. We have

Table II. Summary of numerical results.*

Function	NL	Counter	SA–H (400)	SA–DC (400)	RS–DC (40)	RSG–DC (20)
GP	4	ls	3.2	2.8	5.2	2.6
		lnf/tnf	3/4	2/3	3/4	2/4
		tfe	744	455	193	91
CB3	3	ls	4.6	5.2	4.8	3.2
		lnf/tnf	3/3	2.8/3	3/3	2.4/3
		tfe	758	445	92	51
CB6	6	ls	6	5.2	5.4	3.2
		lnf/tnf	4/5	3.8/4	4.2/6	2.6/4
		tfe	757	455	94	52
BR	3	ls	4	4.8	5.2	3
		lnf/tnf	3/3	3/3	3/3	2/3
		tfe	742	437	88	44
HM	4	ls	4	4.4	4.8	3.2
		lnf/tnf	3.8/4	3.6/4	3.6/4	3.2/4
		tfe	716	451	100	60
H3	3	ls	3.2	7.4	5.2	2.6
		lnf/tnf	2.8/3	2.6/3	2.4/3	2/3
		tfe	832	513	124	64
H6	2	ls	4.6	6.8	6.2	2.4
		lnf/tnf	2/2	1.8/2	2/2	1.4/2
		tfe	1276	600	224	88
S5	5	ls	1.6	6.8	10	2.6
		lnf/tnf	1.4/3	3.8/5	4.6/5	1.8/3
		tfe	1027	573	344	103
S7	7	ls	2.6	8.6	10	2.6
		lnf/tnf	2/6	4.6/6	5.4/7	1.8/4
		tfe	1060	630	377	105
S10	10	ls	3.4	10	10.2	2.8
		lnf/tnf	2.6/6	6/10	5/9	2.2/4
		tfe	1097	655	379	123
KL	Unknown	ls	4.8	6.8	6.8	2.2
		lnf/tnf	3.6/12	5.6/14	7/10	1.4/3
		tfe	1190	708	312	240
EV	Unknown	ls	5.6	6.4	7.8	2.6
		lnf/tnf	5.2/11	6.4/13	5.4/12	2/6
		tfe	1218	510	191	72
ST5	5	ls	NA	6.8	7.8	1.8
		lnf/tnf		3.5/5	3.8/5	1.4/3
		tfe		600	272	92

Table II. *Continued.*

ST7	7	ls	NA	10.6	10.8	2.2
		lnf/tnf		4.2/6	4/5	1.8/4
		tfe		736	378	92
ST10	10	ls	NA	12.4	10	2.2
		lnf/tnf		4.6/7	4.4/8	2/4
		tfe		835	396	77

* *Note*: NL = number of local minima; ls = number of local searches; lnf = average number of local minima found; tnf = total number of minima found in five independent runs; tfe = total number of function evaluations, including function evaluations in sampling, Hessian estimation and local searches; SA–H = H-based method with SA (400 sample points), SA–DC = direct clustering method with SA (400 sample points), RS–DC = direct clustering method with random search (40 sample points), RSG–DC = direct clustering method with random search aimed for the global minima only (20 sample points); NA = Not applicable/available.

experimented with 40 sample points (the same sample size as used in domain elimination [8]) and the results are summarized in column 6 of Table II. With a small sample size of 40, sample points tend to be scattered, which makes Hessian estimation less likely to be useful. Thus, only the direct clustering results are shown. Comparing columns 4, 5 and 6 shows that on average the direct clustering method with random search has performed well with similar effectiveness (in terms of local minima found) and with fewer function evaluations compared to both the H-based and direct clustering methods using one-temperature SA.

It should also be noted that when the region of attraction is identified, the local search procedure plays an important role in whether the corresponding local minimum can be found. The local search procedure may have a tendency to jump to a better minimum even if it starts from the region of attraction of a poorer minimum and is sensitive to the starting point. As can be seen, for most test problems, more regions of attraction than local minima were identified.

If only the global minimum is of interest, the sample size can be further decreased. Column 7 of Table II shows the results for direct clustering with a sample size of only 20. The effectiveness in terms of local minima found is somewhat reduced. But RSG-DC successfully located the global minimum for all but one test problem with a considerable reduction in function evaluations.

7. COMPARATIVE STUDIES

In general, it is very difficult to compare the performance of different global optimization algorithms because the algorithms may have different goals, termination criteria, test problems and ways of reporting results. When reporting numerical results, most researchers have tended to be concerned with only the number of function evaluations and excluded the number of gradient evaluations even though analytical or numerical gradients were used in sample preprocessing [6] or in local search procedures [5, 6, 8].

Tables III and IV present the results of comparative studies considering previous publications. All of these results are the average of five independent runs except the third column of Table IV where both the average and total number (average/total) of local minima found in five runs are reported. When conducting the comparative study, the number of gradient evaluations in Hessian matrix estimation is presented in terms of equivalent function evaluations,

Table III. Comparative study I.*

Function	NL	Counter	SA–H (400)	SA–DC (400)	RS–DC (40)	VQ† (1000)	MLSL (1000)	MLMA (1000)
GP	4	ls	3	3	5	4	3	5
		lnf	3	2	3	3	3	3
		tfe	744	455	193	1068	1091	1117
BR	3	ls	4	5	5	NA	3	3
		lnf	3	3	3		3	3
		tfe	742	437	88		1065	1063
H3	3	ls	3	7	5	2	4	3
		lnf	3	3	2	2	2	2
		tfe	832	513	124	1034	1112	1106
H6	2	ls	5	7	6	3	10	5
		lnf	2	2	2	2	2	2
		tfe	1276	600	224	1149	1986	1454
S5	5	ls	2	7	10	5	5	5
		lnf	1	4	5	5	5	4
		tfe	1027	573	344	1179	1211	1214
S7	7	ls	3	9	10	6	6	5
		lnf	2	5	5	6	6	5
		tfe	1060	630	377	1192	1281	1224
S10	10	ls	3	10	10	7	8	5
		lnf	3	6	5	7	8	5
		tfe	1097	655	379	1198	1346	1238

* *Note*: VQ = VQ-multi-start with clustering [6] (1000 sample points), MLSL = multi-level single linkage [5] (1000 sample points), MLMA = multi-level mode analysis [5] (1000 sample points), NA = Not applicable/available.
† The VQ multi-start with clustering method used the gradient in preprocessing the sample points but the number of the gradient calculations was not reported.

but the number of gradient evaluations in local searches is excluded in order to compare the algorithms in this work on the same basis as the numerical data shown in other researchers' presentations.

In Table III, variations of strategies studied in this work are compared with three cell-based clustering algorithms—VQ [6], MLSL [5], and MLMA [5]. It is noticeable that a larger sample size is shown for the cell-based algorithms. This is required by the use of cells and the Bayesian stopping rules.

The ISO-OCT employed in this study is relatively insensitive to the sample size. Bayesian stopping rules are not needed in our strategy, as the program stops when all local searches terminate. Thus, the direct clustering strategy with simple random search can use a small sample size (40). However, for the H-based strategy with one-temperature SA, if the sample size is too small (<100), then the accepted sample points tend to be scattered, which would result in inaccurate Hessian estimation, making the strategy ineffective. If the sample size is too large (>1000), Hessian estimation would consume a large portion of the function evaluations and make the strategy inefficient.

Table IV. Comparative study II.

Function	NL	Counter	SA–H (400)	SA–DC (400)	RS–DC (40)	DE* (40)	ZM* (40)
CB6	6	ls	6	5	5	4	4
		lnf/tnf	4/5	4/4	4/6	NA/2	NA/2
		tfe	757	455	94	127	316
KL	Unknown	ls	5	7	7	16	2
		lnf/tnf	4/12	6/14	7/10	NA/8	NA/1
		tfe	1190	708	312	5886	5725
EV	Unknown	ls	6	6	8	19	10
		lnf/tnf	5/11	6/13	5/12	NA/11	NA/6
		tfe	1218	510	191	2703	3452

*The number of local minima found reported in zooming and domain elimination are the total number found in five independent runs, while the number of function evaluations is the average of five independent runs. DE = Domain elimination [8]; ZM = Zooming [8].

Table V. Methods to be compared.

Method	Name	Reference
A	Multi-start	Rinnooy Kan and Timmer [19]
B	Controlled random search	Price [2]
C	Density cluster	Torn [4]
D	Clustering with distribution function	De Biase and Frontini [2]
E	Multi-level single linkage	Rinnooy Kan and Timmer [5]
F	Simulated annealing	Dekker and Aarts [20]
G	Simulated annealing based on differential equations	Aluffi-Pentini et al. [21]
H	Hybrid genetic algorithm	Hussain and Al-Sultan [22]
I	Integral global optimization	Zheng and Zhuang [23]
J	DC with simple random search (RSG–DC)	This work

In Table IV, the multi-start clustering strategies are compared with the domain elimination (DE) and zooming (ZM) methods. Both are multi-start methods, but neither employs a clustering algorithm.

As shown in Tables III and IV, the variations of strategies studied in this work appear quite competitive or favourable in comparison to other multi-start (with or without clustering) techniques in terms of the number of local searches performed, the number of minima found, whether the global minimum is located and the number of the function evaluations required.

The comparative study of Table VI is for algorithms intended for only global minima. Table V lists the methods considered and Table VI shows the number of function evaluations reported by different methods. Since most of these methods do not use gradient information, for the purpose of the comparison, we count all gradient evaluations we have used in terms of function evaluations. It can be seen from Table VI that the direct clustering method, with a simple random search, consistently shows fewer required function evaluations in locating the global minimum compared to other methods shown with the exception of one test problem (GP).

Table VI. Number of function evaluations reported by different methods.

Function	GP	BR	H3	H6	S5	S7	S10
Dimension	2	2	3	6	4	4	4
A	4400	1600	2500	6000	6500	9300	11000
B	2500	1800	2400	7600	3800	4900	4400
C	2499	1558	2584	3447	3649	3606	3874
D	378	597	732	807	620	788	1160
E	148	206	197	487	404	432	564
F	563	505	1459	4648	365	558	797
G	5439	2700	3416	3975	2446	4759	4741
H	146	199	191	482	403	521	559
I	1051	1267	1150	3345	2453	3028	2735
J	151	88	127	274	315	313	383
	(91+60*)	(44+40*)	(64+63*)	(88+186*)	(103+212*)	(105+208*)	(123+260*)

*This number is the number of equivalent function evaluations for computing gradient in local searches.

8. CONCLUSIONS

In this paper, we have studied variants of the multi-start with clustering method. The multi-start approach has been one of the most well-known two-phase (global/local) stochastic methods. The random search used in the global phase offers an asymptotic guarantee to ensure the reliability of the method [24] while the local search used in the local phase increases the efficiency of the method. The random search phase can be extended as necessary to provide confidence in convergence for specific applications.

The algorithm developed here can be very efficient for locating the global minimum as shown in Table VI where direct clustering with simple random search (RSG-DC, 20 sample points) performed quite favourably. For identifying multiple minima, direct clustering with simple random search (RS-DC, 40 sample points) also performed well with good results for efficiency and robustness in comparison with existing methods as indicated by Tables III and IV. In comparison to other clustering algorithms, ISO-OCT does not require a large sample size, is less sensitive to the dimension of the problem and identifies clusters properly when they are reasonably formed.

Figure 2 shows that Hessian information can be useful in cluster formation. The results of Table II show that this is especially helpful in identifying regions of attraction of local minima, which leads to robustness in locating multiple local minima. For problems with a few local/global minima and relatively large regions of attraction (common in engineering design), the H-based approach with simulated annealing (SA-H) is more effective and robust with reasonable efficiency as indicated in Table II. The Hessian calculation may require additional function evaluations. These have been controlled here by using an SR1 approach for Hessian estimation but can still be a limiting factor, in terms of computational effort, at a very fine sampling scale.

Overall, the use of Hessian matrix information tends to enhance robustness in identifying multiple local minima for well-behaved problems. When there are many minima or the region of attraction is very small, the H-based strategy becomes less effective. The direct clustering strategy with random search is attractive in comparison to other multi-start techniques in

terms of the number of local searches performed, local minima found and required function evaluations. With substantial modification, both the H-based strategy and direct clustering have been extended to non-linear constrained global optimization problems [25].

ACKNOWLEDGEMENTS

Support of this work under the Dr Drescher grant by the State of New York/UUP Affirmative Action Committee and Computing and Information Technology of the State University of New York at Buffalo is gratefully acknowledged.

REFERENCES

1. Vavasis S. Complexity issues in global optimization: a survey. In *Handbook of Global Optimization*, Horst R, Pardalos PM (eds). Kluwer Academic Publishers: Dordrecht/Boston/London, 1995; 27–41.
2. Dixon LCW, Szego GP (eds). *Towards Global Optimization*. North-Holland: Amsterdam, 1975.
3. Dixon LCW, Szego GP (eds). *Towards Global Optimization II*. North-Holland: Amsterdam, 1978.
4. Törn AA. A search clustering approach to global optimization. In *Towards Global Optimization*, Dixon LCW, Szego GP (eds). North-Holland: Amsterdam, 1978; 49–62.
5. Rinnooy Kan AHG, Timmer GT. Stochastic global optimization methods. Part II: multi level methods. *Mathematical Programming* 1987; **39**:57–78.
6. Jain P, Agogino AM. Global optimization using multistart method. *Advances in Design Automation*, ASME 1989; **DE-19-2**:39–44.
7. Locatelli M. Relaxing the assumptions of the multilevel single linkage algorithm. *Journal of Global Optimization* 1998; **13**:25–42.
8. Elwakeil OA, Arora JS. Two algorithms for global optimization of general NLP problems. *International Journal for Numerical Methods in Engineering* 1996; **39**:3305–3325.
9. Sepulveda AE, Epstein L. The repulsion algorithm, a new multistart method for global optimization. *Structural Optimization* 1996; **11**:145–152.
10. Cha JZ, Mayne RW. The symmetric rank one formula and its application in discrete nonlinear optimization. *Transactions of ASME* 1991; **113**:312–317.
11. Beltracchi TJ, Gabriele GA. A hybrid variable metric update for the recursive quadratic programming method. *Journal of Mechanical Design* 1991; **113**:280–285.
12. Corana A, Marchesi M, Martini C, Ridella S. Minimizing multimodal functions of continuous variables with the 'Simulated Annealing' algorithm. *ACM Transactions on Mathematical Software* 1987; **13**:262–280.
13. Bow Sing-Tze. *Pattern Recognition and Image Preprocessing*. Marcel Dekker Inc.: New York, 1992.
14. Davies DL, Bouldin DW. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1979; **PAMI-1**(2):224–227.
15. Hsueh Y. A program package for global design optimization using the 'Simulated Annealing' algorithm. *Master Thesis*, Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, 1991.
16. *IMSL Math Library*. Visual Numerics, Inc. 1994.
17. Himmelblau DM. *Applied Nonlinear Programming*. McGraw-Hill Book Company: New York, 1972.
18. Tu W. Strategies for global optimization including engineering applications. *Ph.D. Dissertation*, Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, 1999.
19. Rinnooy Kan AHG, Timmer GT. Stochastic methods for global optimization. *American Journal of Mathematical and Management Science* 1984; **4**:7–40.
20. Dekkers A, Aarts E. Global optimization and simulated annealing. *Mathematical Programming* 1991; **50**: 367–393.
21. Aluffi-Pentini F, Parisi V, Zirilli F. Global optimization and stochastic differential equations. *Journal of Optimization Theory and Applications* 1985; **47**:1–16.
22. Hussain MF, Al-Sultan KS. A hybrid genetic algorithm for nonconvex function minimization. *Journal of Global Optimization* 1997; **11**:313–324.
23. Zheng Q, Zhuang DM. Integral global optimization: algorithms, implementations and numerical tests. *Journal of Global Optimization* 1995; **7**:421–454.
24. Jain P. A vector quantization multistart method for global optimization. *Ph.D. Dissertation*, University of California, Berkeley, 1989.
25. Tu W, Mayne RW. An approach to multi-start clustering for global optimization with non-linear constraints. *International Journal for Numerical Methods in Engineering* 2002; **53**:2253–2269.

