Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Kalyanmoy Deb
deb@iitk.ac.in
Department of Electrical and Computer Engineering, Michigan State University, East Lansing, Michigan 48824, USA; Kanpur Genetic Algorithms Laboratory, Department of Mechanical Engineering, Indian Institute of Technology Kanpur, PIN 208 016, India; Department of Information and Service Economy, Aalto University School of Economics, Helsinki, Finland

Amit Saha
amitsaha.in@gmail.com
Kanpur Genetic Algorithms Laboratory, Indian Institute of Technology Kanpur, PIN 208 016, India

Abstract
In a multimodal optimization task, the main purpose is to find multiple optimal solutions (global and local), so that the user can have better knowledge about different optimal solutions in the search space and as and when needed, the current solution may be switched to another suitable optimum solution. To this end, evolutionary optimization algorithms (EA) stand as viable methodologies mainly due to their ability to find and capture multiple solutions within a population in a single simulation run. With the preselection method suggested in 1970, there has been a steady suggestion of new algorithms. Most of these methodologies employed a niching scheme in an existing single-objective evolutionary algorithm framework so that similar solutions in a population are deemphasized in order to focus and maintain multiple distant yet near-optimal solutions. In this paper, we use a completely different strategy in which the single-objective multimodal optimization problem is converted into a suitable bi-objective optimization problem so that all optimal solutions become members of the resulting weak Pareto-optimal set. With the modified definitions of domination and different formulations of an artificially created additional objective function, we present successful results on problems with as large as 500 optima. Most past multimodal EA studies considered problems having only a few variables. In this paper, we have solved up to 16-variable test problems having as many as 48 optimal solutions and for the first time suggested multimodal constrained test problems which are scalable in terms of number of optima, constraints, and variables. The concept of using bi-objective optimization for solving single-objective multimodal optimization problems seems novel and interesting, and more importantly opens up further avenues for research and application.

Keywords
Multimodal optimization, bi-objective optimization, NSGA-II, Hooke-Jeeves exploratory search, multimodal constrained optimization.

1 Introduction
Single-objective optimization problems are usually solved for finding a single optimal solution, despite the existence of multiple optima in the search space. In the presence of multiple global and local optimal solutions in a problem, an algorithm is usually
K. Deb and A. Saha

preferred if it is able to avoid local optimal solutions and locate the true global optimum.

However, in many practical optimization problems having multiple optima, it is wise to find as many such optima as possible for a number of reasons. First, an optimal solution currently favorable (say, due to availability of some critical resources or satisfaction of some codal principles, or others) may not remain so in the future. This would then demand the user to operate at a different solution when such a predicament occurs. With the knowledge of another optimal solution for the problem which is favorable to the changed scenario, the user can simply switch to this new optimal solution. Second, the sheer knowledge of multiple optimal solutions in the search space may provide useful insights into the properties of optimal solutions of the problem. A similarity analysis of multiple optimal (high-performing) solutions may bring about useful innovative and hidden principles, similar to that observed in Pareto-optimal solutions in a multi-objective problem solving task (Deb and Srinivasan, 2006).

Thus, in a multimodal optimization task, the goal is to find multiple optimal solutions and not just one single optimum, as is done in a typical optimization study. If a point-by-point optimization approach is used for this task, the approach should be applied many times, every time hoping to find a different optimal solution. The rise of multimodal optimization research in the evolutionary computation (EC) field has taken place mainly due to the population approach. By working with a number of population members in each generation, EC algorithms facilitate finding and maintaining multiple optimal solutions from one generation to the next. EC researchers employed an additional niching operation for this purpose. Instead of comparing population members randomly chosen from the population in their selection operator, a niching method restricts comparison of solutions to the ones which are similar to each other so that distant good solutions (in the decision variable space) are emphasized. Such efforts date from 1970 with the concept of preselection (Cavicchio, 1970). Thereafter, crowding, sharing, clearing, and many other concepts were suggested to implement the above niching idea.

In the recent past, evolutionary multi-objective optimization (EMO) methods have been used to solve single-objective or other problems, mainly to exploit the diversity preserving property of EMO algorithms. For this purpose, EMO methodology has been used for solving single-objective constrained optimization problems (Surry et al., 1995; Osyczka et al., 2000; Coello and Montes, 2002; Klamroth and Jørgen, 2007), for reducing bloat in genetic programming (De Jong et al., 2001; Bleuler et al., 2001), to aid in solving difficult single-objective optimization problems (Knowles et al., 2001; Jensen, 2006, 2004; Handl et al., 2008), for solving combinatorial optimization problems (Neumann and Wegener, 2006; Jähne et al., 2009), and others. A recent book (Knowles et al., 2008) reports many other multi-objectivization studies.

In this paper, we use the principle of bi-objective optimization to solve single-objective multimodal problems. In addition to the underlying single-objective function, we introduce an additional second objective, the optimum of which will ensure a unique property followed by every optimal solution in the search space which is not shared by any other solution in the search space. Theoretically speaking, one simple strategy would be to use the norm of the gradient vector as the second objective. Since all optimal solutions (global or local) have a zero gradient vector, thereby causing the second objective to take its minimum value, the minimization of the original objective function and minimization of the norm of the gradient vector would make only the optimal solutions lie on the weak Pareto-optimal set of the resulting bi-objective problem. We then suggest a modified EMO procedure to find weak Pareto-optimal solutions, thereby
finding multiple optimal solutions to the multimodal optimization problem in a single
simulation run.

The bi-objective concept is implemented with a couple of pragmatic approaches
so that the suggested idea is computationally efficient and can also be used in non-
differentiable problems. In one of the approaches, we have borrowed a neighborhood
search scheme from the classical Hooke-Jeeves pattern search algorithm (Reklaitis et al.,
1983) in order to reduce the computational complexity further. In problems varying
from two to 16 variables, having 16 to 500 optima, and having a mix of multiple global
and multiple local optima, we demonstrate the working of the proposed procedure.
Most existing multimodal EC studies used one-variable to five-variable problems. The
success of the proposed procedure to much larger and more complex search spaces
remains a hallmark achievement of this paper.

Despite the long history of solving multimodal problems using EC techniques,
none of the studies suggested any specific algorithms for handling constrained multi-
omodal optimization problems. In this paper, for the first time, we suggest a scalable test
problem in which the number of decision variables, the number of constraints, and the
number of optima can all be varied systematically. A slight modification to the proposed
unconstrained procedure is able to solve constrained problems having as many as 32
minima, all lying on the intersection surface of multiple constraint boundaries.

In the remainder of the paper, we provide in Section 2 a brief description of past
multimodal EC studies. The concept of bi-objective optimization for solving multimodal
problems is described in Section 3. Two gradient-based approaches are presented and
simulation results on a simple problem are shown to illustrate the principle of bi-
objective optimization used here. Thereafter, a number of difficulties in using the gra-
dient approach in practice are mentioned and a more pragmatic neighborhood-based
technique is suggested in Section 4. Simulation results on the modified two-variable
Rastrigin’s functions having as many as 500 optima are shown. The scalability issue
of the neighborhood-based technique is discussed and a new Hooke-Jeeves (H-J) pat-
tern search-based algorithm is introduced with the proposed bi-objective optimization
algorithm in Section 5. The results are shown for up to 16-variable problems having
as many as 48 optima. The results are compared with those of a popular multimodal
optimization procedure. Thereafter, a scalable constrained test problem generator for
multimodal optimization is introduced in Section 6. The results for up to 10-variable
problems and having 32 optima are presented. Finally, conclusions and a few extensions
to this study are highlighted in Section 7.

2 Evolutionary Multimodal Optimization

As the name suggests, a multimodal optimization problem has multiple optimum
solutions, of which some can be global optimum solutions having identical objective
function value and some can be local optimum solutions having different objective
function value. Multimodality in a search and optimization algorithm usually causes
difficulty to any optimization algorithm, since there are many attractors to which the
algorithm can become directed. The task in a multimodal optimization algorithm is to
find the multiple optima (global and local) either simultaneously or one after another
systematically. This means that, in addition to finding the global optimum solution(s),
we are also interested in finding a number of other local optimum solutions. Such
information is useful to design-engineers and practitioners for choosing an alternate
optimum solution as and when required.
Evolutionary algorithms (EAs) with some changes in the basic framework have been found to be particularly useful in finding multiple optimal solutions simultaneously, simply due to their population approach and the flexibility in modifying their search operators. For making these algorithms suitable for solving multimodal optimization problems, the main challenge has been to maintain an adequate diversity among population members such that multiple optimum solutions can be found and maintained from one generation to another. For this purpose, niching methodologies are employed in which crowded solutions in the population (usually in the decision variable space) are degraded either by directly reducing the fitness value of neighboring solutions (such as the sharing function approach, Goldberg and Richardson, 1987; Deb and Goldberg, 1989; Mahfoud, 1995; Beasley et al., 1993; Darwen and Yao, 1995) or by directly ignoring crowded neighbors (crowding approach, DeJong, 1975; Mengersheol and Goldberg, 1999; clearing approach, Pétrowski, 1996; Lee et al., 1999; Im et al., 2004; Singh and Deb, 2006; and others; the clustering approach, Yin and Germay, 1993; Streichert et al., 2003; or by other means, Bessaou et al., 2000; Li et al., 2002). A number of recent studies point to the importance of multimodal evolutionary optimization (Singh and Deb, 2006; Deb and Kumar, 1995; Deb et al., 1993). A number of survey and seminal studies summarize such niching-based evolutionary algorithms to date (Rönkkönen, 2009; Deb, 2001). The niching methods are also used for other nongradient methods such as particle swarm optimization (Barrera and Coello, 2009; Zhang and Li, 2005; Parsopoulos and Vrahatis, 2004; Kennedy, 2000; Parrott and Li, 2006), differential evolution (Hendershot, 2004; Li, 2005; Rönkkönen and Lampinen, 2007), and evolution strategies (Shir and Bäck, 2005, 2006). In most of these studies, the selection and/or fitness assignment schemes in a basic EA are modified. In some mating-restriction-based EA studies, recombination between similar solutions is also enforced so as to produce fewer undesirable solutions that lie in nonoptimal regions. The inclusion of a mating restriction scheme to a niching-based EA has been found to improve the online performance of the procedure (Deb, 1989).

In this study, we take a completely different approach and employ a bi-objective optimization strategy using evolutionary algorithms to find multiple optimal solutions simultaneously. We describe the approach in the following section by first using the gradients of the objective function and then using neighboring points.

3 Multimodal Optimization Using Bi-Objective Optimization

Over the past decade, the principle of evolutionary multi-objective optimization methodologies to find multiple trade-off optimal solutions simultaneously in a multi-objective optimization problem has been extensively used to solve various problem-solving tasks. A recent book (Knowles et al., 2008) presents many such ideas and demonstrates the efficacy of such procedures.

In a multimodal optimization problem, we are interested in finding multiple optimal solutions in a single execution of an algorithm. In order to use a multi-objective optimization methodology for this purpose, we need to first identify at least a couple of conflicting (or invariant) objectives for which multiple optimal solutions in a multimodal problem become the trade-off optimal solutions to the corresponding multi-objective optimization problem.

Let us consider the multimodal minimization problem shown in Figure 1 having two minima with different function values:

\[
\text{minimize } f(x) = 1 - \exp(-x^2) \sin^2(2\pi x),
\]

subject to \(0 \leq x \leq 1.\) (1)
Figure 1: A bimodal function.

If we order the minima according to ascending objective function value ($f(x)$), the optimal solutions will line up from the global minimum to the worst local minimum point. In order to have all the minimum points on the trade-off front of a two-objective optimization problem, we need another objective which either conflicts with $f(x)$ (so they appear on a Pareto-optimal front) or is invariant for all minimum points (hence they form a weak Pareto-optimal front). We first suggest a gradient-based method and then present two neighborhood-based approaches.

3.1 Gradient-Based Approach

One property which all minimum points will have in common and which is not shared by other points in the search space (except the maximum and saddle points) is that the derivative of the objective function ($f'(x)$) is zero at these points. We discuss a procedure for distinguishing maximum points from the minimum points later, but first we explain the bi-objective concept here. Let us consider the following two objectives:

\[
\begin{align*}
\text{minimize} & \quad f_1(x) = f(x), \\
\text{minimize} & \quad f_2(x) = |f'(x)|.
\end{align*}
\]

(2)

in the range $0 \leq x \leq 1$, we observe that the minimum (C and G) and maximum points (A, E, and I) of $f(x)$ will correspond to the weak Pareto-optimal points of the above bi-objective problem. Figure 2 shows the corresponding two-dimensional objective space ($f_1$-$f_2$). Any point $x$ maps only on the $f_1$-$f_2$ curve shown in the figure. It is interesting to observe how the $f'$ and $f$ combination makes two different minimum points (C and G) as weak Pareto-optimal solutions of the above bi-objective problem. It is worth reiterating that on the $f_2 = 0$ line, there does not exist any feasible point other than the three points (two corresponding to minimum points and one corresponding to the maximum points) shown in the figure. This observation motivates us to use an EMO procedure in finding all weak Pareto-optimal solutions simultaneously. As a result of the process, we would discover multiple minimum points in a single run of the EMO procedure.

\[\text{We have avoided the cases with saddle points in this gradient approach here, but our later approaches distinguish minimum points from maximum and saddle points.}\]
Figure 2: The objective space of the bi-objective problem of minimizing $f(x)$ and $|f'(x)|$.

Figure 3: The modified objective space with second-order derivative information makes the maximum points dominated.

One drawback of the above procedure is that it will also make all maximum points (A, E, and I) as weak Pareto-optimal along with the minimum points. To avoid this scenario, we suggest an alternate second objective, as follows:

$$\begin{align*}
\text{minimize} & \quad f_1(x) = f(x), \\
\text{minimize} & \quad f_2(x) = |f'(x)| + (1 - \text{sign}(f''(x))) ,
\end{align*}$$

where sign() returns $+1$ if the operand is positive and $-1$ if the operand is negative. Thus, for a minimum point, $f''(x) > 0$, the second term in $f_2(x)$ is zero and $f_2(x) = |f'(x)|$. On the other hand, for a maximum point, $f''(x) < 0$, and $f_2(x) = 2 + |f'(x)|$. For saddle points having $f''(x) = 0$, $f_2(x) = 1 + |f'(x)|$. This modification will make the maximum and $f''(x) = 0$ points dominated by the minimum points. Figure 3 shows the objective space with this modified second objective function on the same problem considered in Figure 1.
3.2 Modified NSGA-II Procedure

The above discussion reveals that the multiple minimum points become different weak Pareto-optimal points of the corresponding bi-objective minimization problem given in Equations (2) and (3). However, state of the art EMO algorithms are usually designed to find Pareto-optimal solutions and are not expected to find weak Pareto-optimal solutions. For our purpose here, we need to modify an EMO procedure to find weak Pareto-optimal points. Here, we discuss the modifications made on a specific EMO algorithm (the NSGA-II procedure [Deb et al., 2002]) to find weak Pareto-optimal solutions:

1. First, we change the definition of domination between two points $a$ and $b$. Solution $a$ dominates solution $b$, if $f_2(a) < f_2(b)$ and $f_1(a) \leq f_1(b)$. Thus, if two solutions have identical $f_2$ value, they cannot dominate each other. This property will allow two solutions having identical $f_2$ values to co-survive in the population, thereby allowing us to maintain multiple weak Pareto-optimal solutions, if found in an EMO population.

2. Second, we introduce a rank-degrading concept around some selected nondominated points in the objective space, so as to avoid crowding around minimum points. For this purpose, all points of a nondominated front (using the above modified domination principle) are first sorted in ascending order of $f(x)$. Thereafter, the point with the smallest $z = f(x)$ solution (having objective values $(f_1, f_2) = (z, z')$) is kept and all population members (including the current nondominated front members) in the neighborhood around a box ($f_1 \in (z, z + \delta_f)$ and $f_2 \in (z' + \delta'_f)$) are degraded and assigned a large nondomination rank. The next solution from the sorted $f(x)$ list is then considered and all solutions around its neighborhood are degraded likewise. After all nondominated front members are considered, the remaining population members are used to find the next nondominated set of solutions and the above procedure is repeated. This process continues until all population members are either degraded or assigned a nondomination level. All degraded members are accumulated in the final front and solutions with a larger crowding distance value based on $f(x)$ are preferred.

These two modifications ensure that weak nondominated solutions with identical $f_2(x)$ values but different $f(x)$ values are emphasized in the population and nondominated solutions with smaller $f(x)$ values are preferred and solutions around them are deemphasized.

3.3 Proof-of-Principle Results with the Gradient Approach

In this subsection, we show the results of the modified NSGA-II procedure on a single-variable problem having five minima:

$$f(x) = 1.1 - \exp(-2x) \sin^2(5\pi x), \quad 0 \leq x \leq 1.$$  \hspace{1cm} (4)

The following parameter values are used: population size = 60, SBX probability = 0.9, SBX index = 10, polynomial mutation probability = 0.5, mutation index = 50, $\delta_f = 0.02$, $\delta'_f = 0.1$, and maximum number of generations = 100.

Figure 4 shows that the modified NSGA-II is able to find all five minima of this problem. Despite having sparse points near all five minima, the procedure with its modified domination principle and the rank-degrading approach is able to maintain a well-distributed set of points on all five minima in a single simulation.
In addition to finding all the minimum points, the procedure also finds three of the six maximum points (points A, B, and C, shown in Figure 4). Since the second objective is the absolute value of $f'(x)$, for both minimum and maximum points, it is zero. Thus, all minimum and maximum points are weak dominated points and become the target of the above procedure. However, the consideration of minimization of $f(x)$ emphasizes the minimum points more than the maximum points.

In order to avoid finding the maximum points completely, next, we use the modified second objective function involving the second derivative $f''(x)$, given in Equation (3). Identical parameter settings to that in the above simulation are used here. Since the maximum points now get dominated by any minimum point, this time, we are able to cleanly find all the five minima alone (see Figure 5). It is also interesting to note that due to the emphasis on finding the weak Pareto-optimal solutions in the suggested
bi-objective approach, some solutions near the optima are found. Since gradient information is used in the second objective, undesired solutions (in nonoptimal regions) get dominated by minimum solutions and there is no need for an explicit mating restriction procedure here, which is otherwise recommended in a typical evolutionary multimodal optimization algorithm (Deb and Goldberg, 1989).

3.4 Practical Difficulties with Gradient-Based Methods

Although the concept of first-order and second-order gradient information allows all minimum points to have a common property which enabled us to find multiple minimum points using a bi-objective procedure, the gradient-based approaches have certain well-known difficulties.

First, such a method has restricted applications, because the method cannot be applied in problems which do not have a gradient at some intermediate points or at the minimum points. The gradient information drives the search toward the minimum points and if gradient information is not accurate, such a method is not likely to work well.

Second, the gradient information is specific to the objective function. The range of gradient values near one optimum point may vary from another optimum point, depending on the flatness of the function values near the optimum points. A problem having a differing gradient value near different optima will provide unequal importance to different optima, thereby causing an algorithm to have difficulty in finding all optimum points in a single run.

A third and crucial issue relates to the computational complexity of the approach. Although the first-order derivative approach requires $2n$ function evaluations for each population member, this comes with the additional burden of finding the maximum points. To avoid finding unwanted maximum points, the second-derivative approach can be used. But the computation of all second derivatives numerically takes $2n^2 + 1$ function evaluations (Deb, 1995), which is prohibitory for large-sized problems.

To illustrate the above difficulties, we consider a five-variable modified Rastrigin’s function having eight minimum points (the Rastrigin function is described later in Equation (6) with $k_1 = k_2 = k_3 = 1, k_4 = 2$, and $k_5 = 4$). The norm of the gradient vector ($\| \nabla f(x) \|$) is used as the second objective. When we apply our proposed gradient-based approach with the identical parameter setting as above to solve this problem, out of eight minimum points, only two are found in the best of 10 runs. Figure 6 shows the objective space enumerated with about 400,000 points created by equispaced points in each variable within $x_i \in [0, 1]$. The eight minimum points are shown on the $f_2 = 0$ line and paths leading to these minimum points are obtained by varying $x_5$ and by fixing $x_1$ to $x_4$ to their theoretical optimal values. The obtained two minimum points are shown with a diamond. It is interesting to note from the figure that points near the minimum solutions are scarce. On the other hand, there are many maximum points lying on the $f_2 = 0$ line, which are accessible from the bulk of the objective space. With the first derivative information alone, an algorithm can be led to these maximum points.

The above study is pedagogical and demonstrates that a multimodal optimization problem can in principle be converted to an equivalent bi-objective problem by using first-order and second-order derivative-based optimality conditions. However, there are implementational issues which will restrict the use of the gradient-based methods to reasonably higher-dimensional problems. Nonetheless, the idea of using a bi-objective optimization technique to find multiple optimum points in a multimodal optimization problem is interesting and next we suggest a more pragmatic approach.
K. Deb and A. Saha

Figure 6: The two-objective problem is illustrated for the five-variable modified Rast-rigin’s function. Variation of $x_5$ by fixing all other variables to their optimal values show eight paths (two almost coincide having almost equal function value) culminating in eight multimodal minimum points.

Figure 7: The second objective is obtained by counting the number of better neighboring solutions. The minimum solution will correspond to $f_2 = 0$.

4 Multimodal Optimization Using Neighboring Solutions

Instead of checking the gradients for establishing optimality of a solution ($x$), we can simply compare a sample of neighboring solutions with the current solution $x$. The second objective function $f_2(x)$ can be assigned as the count of the number of neighboring solutions that are better than the current solution $x$ in terms of their objective function ($f(x)$) values. Figure 7 illustrates the idea on a single-variable hypothetical problem. Point A is the minimum point of this function and both neighboring solutions are worse
than point A, thereby making $f_2(A) = 0$. On the other hand, any other point (which is not a minimum point) will have at least one neighboring point better than that point. For example, point B has one better neighboring solution, thereby making $f_2(B) = 1$, and point C has two better neighboring points, making $f_2(C) = 2$. To implement the idea, a set of $H$ solutions in the neighborhood of $x$ can be computed systematically. One simple approach would be to use the Latin hypercube sampling (LHS) method with $H$ divisions in each dimension and pick a combination of divisions such that a representative from each division in every dimension is present. For a large-dimensional problem and with a reasonably large $H$, this procedure may not provide a good sampling of the neighborhood. Another idea would be to make $k$ divisions (e.g., $k = 2$) along each dimension and sample a point in each division, thereby constructing a total of $H = k^n$ points to compute the second objective. Clearly, other computationally more efficient ideas are possible and we suggest one such approach in a later section.

4.1 Modified NSGA-II with Neighboring Point Count

As mentioned, the second objective $f_2(x)$ is computed by counting the number of neighboring solutions that are better (in terms of the objective function value ($f(x)$)) than the current point $x$. Thus, only solutions close to a minimum solution will have a zero count on $f_2$, as locally there does not exist any neighboring solution smaller than the minimum solution. Since $f_2$ now takes integer values only, the $f_2$-space is discrete.

To introduce the niching idea as before on the objective space, we also consider all solutions having an identical $f_2$ value and rank-degrade all solutions within a distance $\delta_f$ from the minimum $f(x)$ value. Figure 8 illustrates the modified nondomination procedure. A population of 40 solutions are plotted on a $f_1$-$f_2$ space. Three non-dominated fronts are marked in the figure to make the domination principle along with the niching operation clear. The population has one point with $f_2 = 8$, two points with $f_2 = 9$, seven
points with $f_2 = 10$, and so on, as shown in the figure. The point with $f_2 = 8$ dominates (in the usual sense) both $f_2 = 9$ points and the right-most four $f_2 = 10$ points. In a typical scenario, the minimum $f_1$ point among $f_2 = 10$ points would have dominated the other six $f_2 = 10$ points. But due to the modified domination principle, three left-most points with $f_2 = 10$ are nondominated to each other and are nondominated with the $f_2 = 8$ point as well. However, the niching consideration makes the second point on the $f_2 = 10$ line from the left dominated by the first point. This is because the second point is within $\delta_f$ distance from the first point. The third point on the $f_2 = 10$ line is more than $\delta_f$ distance away from the left-most point on the $f_2 = 10$ line and hence the third point qualifies to be on the first nondominated front. Since all points with $f_2 = 11$ get dominated by the left-most point on the $f_2 = 10$ line, none of them qualifies to be on the first nondominated front. However, the left-most point with $f_2 = 12$ is nondominated (in the usual sense) with the other first front members. Thus, the best nondominated front is constituted with one point having $f_2 = 8$, two points having $f_2 = 10$, and one point having $f_2 = 12$. These points are joined with a solid line in the figure to show that they belong to the same front.

Similarly, the second nondominated front members are identified by using the modified domination principle and the niching operation and are marked in Figure 8 with a dashed line. It is interesting to note that front classification causes a different outcome from the typical domination principle would have produced. Points lying sparsely in smaller $f_2$ lines and having smaller $f_1$ value are emphasized. Such an emphasis will eventually lead the EMO procedure to solutions on the $f_2 = 0$ line and the niching procedure will help maintain multiple solutions in the population.

4.2 Proof-of-Principle Results

In this section, we present simulation results of the above algorithm, first on a number of two-variable test problems and then on higher-dimensional problems. We consider the following two-variable problem:

\[
\begin{align*}
\text{minimize} & \quad f(x_1, x_2) = (x_1^2 + x_1 + x_2^2 + 2.1x_2) + \sum_{i=1}^{2} 10(1 - \cos(2\pi x_i)), \\
\text{subject to} & \quad 0.5 \leq x_1 \leq (K_1 + 0.5), \quad 0.5 \leq x_2 \leq (K_2 + 0.5).
\end{align*}
\]

(5)

There is a minimum point close to every integer value of each variable within the lower and upper bounds. Since there are $K_i$ integers for each variable within $x_i \in [0.5, K_i + 0.5]$, the total number of minima in this problem are $M = K_1K_2$. First, we consider $K_1 = K_2 = 4$, so that there are 16 minima. We use a population of size 160 (meaning an average of 10 population members are expected around each minimum point), SBX probability of 0.9, SBX index of 10, polynomial mutation probability of 0.5, mutation index of 20, and run each NSGA-II for 100 generations. The parameter $\delta_f = 0.1$ is used here. To make proof-of-principle runs, four neighboring points are created at random within $\pm 0.1$ of each variable value. Figure 9 shows the outcome of the proposed procedure: All 16 minima are found by the procedure in a single simulation run. Since the best nondominated front members are declared as the outcome of the proposed procedure, the optimum solutions are cleanly found. To have a better understanding of the working principle of the proposed procedure, we record the population average $f_2$ value and the number of minima found in every generation, and plot them in Figure 10. It is interesting to note that on average, the random initial population members have about two (out of four) neighboring points better than them. As generations increase, points with a lesser number of better neighboring points are discovered and eventually
Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Figure 9: All 16 minimum points are found by the proposed procedure.

Figure 10: Variation of population average $f_2$ and the number of minima found with generation counter.

At generation 12, all population members have no better neighboring points. Since neighboring points are created around a finite neighborhood, a point with $f_2 = 0$ does not necessarily mean that the point is a minimum point (it can be a point close to the true minimum point as well), but the emphasis on the minimum $f_1(x)$ provided in the domination principle allows solutions close to the true minimum points to be found in subsequent generations.

At each generation, the number of population members which are within a normalized Euclidean distance of 0.03 from the true minimum in each variable dimension is noted and plotted in the above figures. Interestingly, it took initially three generations for the proposed procedure to find a single solution close to a minimum point and thereafter an increasing number of points are found close to true minimum points with succeeding generations. At generation 12 when all population members have $f_2 = 0$, 12 out of 16 minima are found. But eventually at generation 35, all 16 minimum points are found by NSGA-II operators. We would like to stress the fact that the proposed
modified domination principle and niching operation together are able to emphasize and maintain various minimum points and also able to accommodate an increasing number of minimum points, as and when they are discovered. This is a typical performance of the proposed NSGA-II procedure, which we have observed over multiple runs.

4.3 Scalability Results up to 500-Optima Problems

Staying with the same two-variable problem, we now perform a scalability study in terms of the number of optima present in the problem. We vary the population size as $N = 10m$ (where $m$ is the number of minima) where other parameter values are the same as in the case of the 16-minima problem presented above. We construct a 20-minima problem ($K_1 = 5$, $K_2 = 4$), a 50-minima problem ($K_1 = 10$, $K_2 = 5$), a 100-minima problem ($K_1 = 10$, $K_2 = 10$), a 200-minima problem ($K_1 = 20$, $K_2 = 10$), and a 500-minima problem ($K_1 = 25$, $K_2 = 20$). An optimum is considered to be found if a population member within a normalized Euclidean distance of 0.03 from the true location of the optimum is obtained. Final population members for the 100-minima and the 500-minima problems are shown in Figures 11 and 12, respectively. In the case of the 100-minima problem, 97 minimum points are found; and in the case of 500-minima problem, 483 minimum points are found by the proposed approach.

Figure 13 shows the number of minima found in each of the five problems studied with 10 runs for each problem. The average numbers of obtained minima are shown with circles, and worst and best numbers are shown with bars (which can just barely be seen in Figure 13). The proposed algorithm is found to be reliably scalable in finding up to 500 optimal points in a single simulation. The modified domination and the niching strategy seems to work together in finding almost all available minima (and as large as 500) for the chosen problems.

4.4 Difficulties in Handling a Large Number of Variables

In the above simulations with the proposed bi-objective procedure, two points were created around the current point in each variable dimension to determine the second
Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Figure 12: Four-hundred eighty-three out of 500 minimum points are found by the proposed procedure.

Figure 13: Almost all minimum points are found on two-variable problems by the proposed procedure.

5 A Faster Approach with Hooke-Jeeves Exploratory Search

Instead of generating $H$ solutions at random around a given solution point ($x$), we can choose the points judiciously using an exploratory search performed by the classical Hooke-Jeeves optimization algorithm (Reklaitis et al., 1983). The procedure starts with the first variable ($i = 1$) dimension and creates two extra points $x'_c \pm \delta_{hj}$ around the current solution $x' = x$. Thereafter, three solutions ($x' - \delta_{hj}e_i, x', x' + \delta_{hj}e_i$; where $e_i$ is the unit vector along $i$th variable axis in the $n$-dimensional variable space) are...
Table 1: Optimal $x_i$ values for different $k_i$ values. There are $k_i$ number of solutions for each $k_i$.

<table>
<thead>
<tr>
<th>$k_i$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_i$</td>
<td>0.49498</td>
<td>0.24874</td>
<td>0.16611</td>
<td>0.12468</td>
</tr>
<tr>
<td></td>
<td>0.74622</td>
<td>0.49832</td>
<td>0.37405</td>
<td>0.62342</td>
</tr>
<tr>
<td></td>
<td>0.83053</td>
<td>0.37405</td>
<td>0.87279</td>
<td>0.87279</td>
</tr>
</tbody>
</table>

compared with their function values and the best is chosen. The current solution $x^c$ is then moved to the best solution. Similar operations are done for $i = 2$ and continued for the remaining variables. Every time a solution having a better objective value than the original objective value $f(x)$ is encountered, the second objective value $f_2(x)$ is incremented by one. This procedure requires a total of $2n$ function evaluations to compute $f_2(x)$ for every solution $x$ in solving an $n$-dimensional problem.

Although this procedure requires a similar amount of additional function evaluations ($2n$) to that in a numerical implementation of the first-derivative approach described earlier, the idea here is more generic, avoids second-derivative computation and does not have the difficulties associated with the gradient-based approach.

5.1 Results with the H-J Approach

The past multimodal evolutionary algorithms studies handled problems having few variables. In this section, we present results on a number of multi-dimensional multimodal test problems. First, we define a scalable $n$-variable test problem, as follows:

\[
MMP(n) : \minimize f(x) = \sum_{i=1}^{n} 10(1 + \cos(2\pi k_i x_i)) + 2k_i x_i^2, \\
\text{subject to } 0 \leq x_i \leq 1, \quad i = 1, 2, \ldots, n.
\]

(6)

This function is similar to Rastrigin’s function. Here, the total number of global and local minima is $M = \prod_{i=1}^{n} k_i$. We use the following $k_i$ values for the three MMP problems, each having 48 minimum points:

MMP(4): $k_1 = 2, k_2 = 2, k_3 = 3, k_4 = 4$.

MMP(8): $k_2 = 2, k_4 = 2, k_6 = 3, k_8 = 4, k_i = 1, \text{ for } i = 1, 3, 5, 7$.

MMP(16): $k_4 = 2, k_8 = 2, k_{12} = 3, k_{16} = 4, k_i = 1, \text{ for } i = 1−3, 5−7, 9−11, 13−15$.

All of these problems have one global minimum and 47 local minimum points. The theoretical minimum variable values are computed by using the first-order and second-order optimality conditions and solving the resulting root finding problem numerically:

\[
x_i - 5\pi \sin(2\pi k_i x_i) = 0.
\]

In the range $x_i \in [0, 1]$, the above equation has $k_i$ roots. Different $k_i$ values and corresponding $k_i$ numbers of $x_i$ variable values are shown in Table 1. In all runs, we use $\delta_{ij} = 0.005n$. For the domination definition, we have used $\delta_f = 0.5$. For all problems, we have used a population size of $N = 15 \max(n, M)$, where $n$ is the number of variables and $M$ is the number of optima. An optimum is considered to be found if a population member within an Euclidean distance of 0.05 from a true location of the optimum is obtained. Crossover and mutation probabilities and their indices are the same as before.
Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Figure 14: Only 14 of the 48 minimum points are found for the MMP(4) problem by the objective space niching algorithm.

(In Section 5.5, we perform a parametric study and present the results of the proposed algorithm for different values of $\delta_{hj}$ and $\delta_f$.)

First, we solve the four-variable ($n = 4$) problem. Figure 14 shows the theoretical objective value $f(x)$ of all 48 minimum points with diamonds. The figure also shows the minimum points found by the proposed H-J-based NSGA-II procedure with circles. Only 14 out of 48 minimum points are found. We explain this (apparently poor) performance of the proposed procedure here.

The figure shows that although there are 48 minima, a number of minima have identical objective values. For example, the 14th, 15th, and 16th minimum points have an identical function value of 5.105. Since the modified domination procedure deemphasizes all solutions having a difference of $\delta_f = 0.5$ function value from each other, from a cluster of multiple minimum points having identical function value only one of the minimum points is expected to be found; other minimum solutions will be degraded. This scenario can be observed in Figure 14. Once a point is found, no other minimum point having an identical objective function value is obtained by the algorithm.

5.2 Finding Multiple Equal-Valued Optimal Solutions

In order to find multiple minimum points having identical or almost equal function values (e.g., multiple global minima), we need to perform an additional variable-space niching among closer objective solutions by emphasizing solutions having widely different variable vectors. To implement the idea, we check the normalized Euclidean distance (in the variable space) of any two solutions having function values within $\delta_f$, thereby introducing a new parameter $\delta_x$. If the normalized Euclidean distance between the two solutions is greater than the parameter $\delta_x$, we assign both solutions an identical nondominated rank; otherwise, both solutions are considered to be arising from the same optimal basin and we assign a large dominated rank to the solution having the worse objective value $f(x)$. In all runs of this section, we use $\delta_x = 0.2$.

Figure 15 summarizes the result obtained by this modified procedure. All 48 minimum points are discovered by the modified procedure (with a Euclidean distance of 0.05 from the true optimal locations).
Next, we solve the $n = 8$-variable and 16-variable problems using the modified procedure. Figures 16 and 17 show the corresponding results. The modified procedure is able to find all 48 minimum points on each occasion.

There are a few multimodal problems having 8 or 16 variables that have been attempted in the existing multimodal evolutionary optimization studies. It is noteworthy
5.3 Finding Multiple Global Optimal Solutions

Next, we modify the objective function as follows:

\[ f(x) = \sum_{i=1}^{n} 10 + 9 \cos(2\pi k_i x_i). \]

In this problem, we use \( n = 16 \) and \( k_i \) values are chosen as in MMP(16) above, so that there are 48 minima having an identical objective value of \( f = 16 \). Interestingly, there is no locally minimum solution in this problem. We believe this problem will provide a stiff challenge to the variable space niching procedure, as all 48 global minimum points will lie on an identical point on the objective space and the only way to distinguish them from each other would be to investigate the variable space and emphasize distant solutions.

Figure 18 shows that the proposed niching-based NSGA-II procedure is able to find all 48 minimum points for the above 16-variable modified problem within a Euclidean distance of 0.05 from the true optimal locations. Exactly the same results are obtained in 25 other runs (out of a total of 30 runs) each starting with a different initial population. It is interesting that a simple variable space niching approach described above is adequate to find all 48 minimum points of the problem.

Most multimodal EAs require additional parameters to identify niches around each optimum. Our proposed H-J-based procedure is not free from additional parameters. However, both \( \delta_f \) and \( \delta_x \) parameters directly control the differences between any two optima in objective and decision variable spaces. Thus, the setting of these two parameters can be motivated from a practical standpoint. However, our recent adaptive approach (Saha and Deb, 2010) for fixing the parameters based on population statistics is encouraging and is worth investigating further. Nevertheless, the demonstration of successful working of the proposed approach up to 16-variable problems and having
Figure 18: All 48 global minimum points are found for the modified MMP(16) problem.

as many as 48 optima amply indicates its efficacy and should motivate further studies in favor of the proposed bi-objective concept.

5.4 Comparison with the Clearing Approach

In some earlier studies (Pétrowski, 1996; Singh and Deb, 2006), it was reported that the clearing approach performs better on most occasions compared to other evolutionary multimodal approaches. Here, we test the clearing approach on our test problems, mainly to investigate whether the clearing approach can scale to higher-dimensional problems having a large number of optima. A previous study (Singh and Deb, 2006) applied the clearing approach to the 5-variable to 25-variable hump problem, constructed to have a number of optima each with a specified height and basin of attraction. The study reported that it requires a large population size (1,000 to 3,000) to locate about 66% of the optima, each represented by a solution within 15% of the size of the basin of attraction of the optimum. Here we perform a more stringent application and implement the original clearing approach (Pétrowski, 1996) on a real-parameter genetic algorithm using the stochastic remainder roulette-wheel selection operator (Goldberg, 1989), SBX recombination operator, and the polynomial mutation operator (Deb, 2001). Since the proportionate selection operator is used in the clearing approach, only maximization problems (and problems having non-negative objective values) can be solved. In the clearing approach, population members are first sorted based on their objective function value from best to worst. Then, except the first $\kappa$ sorted members, all subsequent members within a niche radius distance of $\sigma_c$ from the best population member are forced to have an objective value of zero. Thereafter, the next $\kappa$ solutions are retained and all other solutions within $\sigma_c$ are cleared and assigned a value of zero. This process continues until all population members are handled. The stochastic remainder selection operator is then applied on the modified objective values.

We have attempted to solve the MMP(2) (derived from Equation (6)) by converting it into a maximization problem as follows:
Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Figure 19: Optimal points found by clearing GA (triangles) and our approach (circles). Twenty-three out of 48 optima are found by the clearing approach with an accuracy of 0.005, whereas our approach finds all 48 optima with the same accuracy.

\[
\text{MMP}(2): \quad \begin{array}{c}
\text{maximize} \\
\quad f(x) = \sum_{i=1}^{n} (20 + 2k_i) - 10(1 + \cos(2\pi k_i x_i)) + 2k_i x_i^2,
\end{array}
\]

subject to \(0 \leq x_i \leq 1, \quad i = 1, 2, \ldots, n\). (7)

We choose \(k_1 = 6\) and \(k_2 = 8\), so that we have a two-variable, 48-optima problem. After a number of simulations with different \(\kappa\) and \(\sigma_c\) values, we found that \(\kappa = 3\) and \(\sigma_c = 0.12\) worked the best for this problem. To declare an optimum being found, we check if there is any solution in a population close to the 0.005 distance (in the Euclidean space) away from the known optimum. Figure 19 shows the final population members obtained using the clearing GA and our approach with \(N = 500\) for a typical run. Other GA parameters are the same as used in our proposed approach.

To give a better insight on how the clearing procedure performs vis-à-vis our proposed H-J procedure, consider the above 48-optima problem and the two-variable 96-optima problem (obtained with \(k_1 = 8\) and \(k_2 = 12\)). We have used 500 and 1,000 population members for the above two cases, respectively. Figures 20 and 21 show the generation wise mean number of optima obtained over 30 independent runs for the clearing and H-J procedures. As the figures show, the clearing procedure is able to find only about half of the desired number of optima in both the cases, whereas the proposed procedure is able to find all optima. The clearing procedure makes most of the population members unimportant by simply declaring them to have zero fitness. To find a large number of optima, the clearing procedure understandably requires a large population, so that an adequate number of important solutions are present in the population for the search process to lead a parallel search from them and converge to all optima in the search space simultaneously. For solving a multimodal problem having a large number of optima using the clearing approach, the requirement of a large population size seems to be a serious shortcoming.

To test the clearing procedure further on high dimensional problems, next we attempt to solve 4-variable to 16-variable versions of the MMP(\(n\)) problem (after...
converting it into a maximization problem). We tried various combinations of $\kappa$ and $\sigma_c$ values, but we failed to obtain even a single run (out of 30 runs) in which all desired optima are found. Table 2 shows a comparative evaluation of the clearing and our H-J approach in terms of number of optima found, the number of runs that locate all desired optima, and the number of function evaluations needed to find all desired optima. For problems with $n = 2, 4, \text{ and } 8$, we have used a population of size 500 and run up to a maximum of 500 generations of both algorithms. For the 16-variable problem, we have used 1,000 population members and run both algorithms up to a maximum of 1,000 generations. An optimum is said to have been found when the normalized Euclidean distance in the variable space between the exact and the obtained optimum is less than or equal to 0.005 for $n = 2$, and 0.05 for $n = 4, 8, \text{ and } 16$. The number of
Table 2: Comparative results (over 30 independent runs) of the clearing GA procedure and H-J-based procedure on MMP(2), MMP(4), MMP(8), and MMP(16).

<table>
<thead>
<tr>
<th></th>
<th>Number of optima found</th>
<th>Success rate</th>
<th>Number of function evaluations (median, SD)</th>
<th>Objective function error (median, SD)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clearing GA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>24/48</td>
<td>0</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>4</td>
<td>25/48</td>
<td>0</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>8</td>
<td>4/48</td>
<td>0</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>16</td>
<td>2/48</td>
<td>0</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Proposed H-J GA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>48/48</td>
<td>90.00</td>
<td>(1.3520e+06, 3.1802e+05)</td>
<td>(1.1357e–08, 7.6573e–08)</td>
</tr>
<tr>
<td>4</td>
<td>48/48</td>
<td>93.33</td>
<td>(1.9825e+05, 1.3225e+05)</td>
<td>(3.4834e–08, 3.3521e–08)</td>
</tr>
<tr>
<td>8</td>
<td>48/48</td>
<td>93.33</td>
<td>(5.0000e+06, 6.6298e+05)</td>
<td>(4.6700e–07, 4.4205e–07)</td>
</tr>
<tr>
<td>16</td>
<td>48/48</td>
<td>86.67</td>
<td>(1.1818e+07, 5.5476e+06)</td>
<td>(1.7000e–03, 1.6400e–02)</td>
</tr>
</tbody>
</table>

function evaluations is recorded until all expected optima are found with the above termination criterion on the solution accuracy. As mentioned earlier and as is also clear from Table 2, the clearing approach is not able to find all the desired optima for any problem. For two-variable and four-variable problems, about 50% optima are found, but since none of them is able to find 100% of the optima, we do not report any function evaluation count for this approach. It is clear that for higher dimensional problems, the performance of the clearing approach degrades. On the other hand, our proposed H-J approach is able to find all desired optima with a reasonable number of function evaluations in all problems. The final column in Table 2 (objective function error) shows the difference in function value from the exact optimal function value obtained with the function evaluations mentioned in column 5.

5.5 Studying the Effect of Parameters

In the proposed H-J-based procedure, we have made use of three different parameters: $\delta_{hj}$, $\delta_f$, and $\delta_x$. Earlier, we chose the values of these parameters that provided the best performance over a number of trial and error runs. Here, we perform a parametric study to evaluate the sensitivity of our algorithm to these parameters. We use the same termination criterion as that used in the clearing study above.

Table 3 shows the performance measure of the H-J-based algorithm for different values of the three parameters on the eight-variable problem MMP(8) introduced earlier. A successful run is one in which our algorithm is able to get all the 48 optima of the MMP(8) problem. As the results show, the best values of $\delta_{hj}$, $\delta_f$, and $\delta_x$ are found to be 0.1, 10.0, and 0.2, respectively. The algorithm is sensitive to the choice of value of $\delta_x$, since it is directly responsible for preserving multiple optima having an identical function value. At the same time, the algorithm is robust to the choice of $\delta_f$. Our experiments also suggest the same values of the parameters for MMP(4) and MMP(16) problems as well.

Next, we use the above-obtained parameter values and re-solve the 16-variable, 48-optima problem MMP(16). It is interesting to note from Table 4 that with the new parameter setting, the algorithm takes a lesser number of function evaluations to find all the optima. The success rate is also slightly higher. Since the algorithm does not terminate until all desired optima are found, the near-optimal solutions obtained early on in a run keep getting modified as other optima in the search space are being found.
Table 3: Performance of the H-J-based procedure with respect to different values of $\delta_h$, $\delta_x$, and $\delta_f$.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
<th>Success rate</th>
<th>Number of function evaluations (median, SD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta_h$</td>
<td>0.03</td>
<td>66.66</td>
<td>(7.1250e+05, 9.1497e+05)</td>
</tr>
<tr>
<td>($\delta_f = 0.5$, $\delta_x = 0.2$)</td>
<td>0.035</td>
<td>83.33</td>
<td>(4.8750e+05, 9.8799e+05)</td>
</tr>
<tr>
<td>0.04</td>
<td>83.33</td>
<td>(5.0000e+05, 6.6298e+05)</td>
<td></td>
</tr>
<tr>
<td>0.045</td>
<td>96.66</td>
<td>(3.8750e+05, 2.9870e+05)</td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>93.33</td>
<td>(4.0000e+05, 5.8576e+05)</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>100</td>
<td>(1.0187e+06, 3.2819e+05)</td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>0</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>$\delta_f$</td>
<td>0.1</td>
<td>0</td>
<td>—</td>
</tr>
<tr>
<td>($\delta_h = 0.1$, $\delta_x = 0.2$)</td>
<td>0.2</td>
<td>0</td>
<td>—</td>
</tr>
<tr>
<td>0.5</td>
<td>100</td>
<td>(1.0187e+06, 3.2819e+05)</td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>100</td>
<td>(6.6958e+05, 1.6003e+05)</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>100</td>
<td>(6.1250+05, 9.3030e+04)</td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>100</td>
<td>(3.6875+05, 9.5690e+04)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>(3.2500e+05, 3.6760e+04)</td>
<td></td>
</tr>
<tr>
<td>$\delta_x$</td>
<td>0.1</td>
<td>100</td>
<td>(3.3745e+05, 9.8921e+04)</td>
</tr>
<tr>
<td>($\delta_h = 0.1$, $\delta_f = 10.0$)</td>
<td>0.15</td>
<td>100</td>
<td>(3.2500e+05, 1.2547e+05)</td>
</tr>
<tr>
<td>0.2</td>
<td>100</td>
<td>(3.2500e+05, 3.6760e+04)</td>
<td></td>
</tr>
<tr>
<td>0.25</td>
<td>0</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0</td>
<td>—</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Results (over 30 runs) of the H-J-based procedure on MMP(16) reporting the number of function evaluations required to find all the optima and the average error in the obtained optima.

<table>
<thead>
<tr>
<th>$(\delta_h, \delta_f, \delta_x)$</th>
<th>Before parametric study</th>
<th>After parametric study</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.08, 0.5, 0.2)</td>
<td>(0.1, 10.0, 0.2)</td>
<td></td>
</tr>
<tr>
<td>Success rate</td>
<td>86.67</td>
<td>90.0</td>
</tr>
<tr>
<td>Function evaluations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best</td>
<td>5.3802e+06</td>
<td>1.9404e+06</td>
</tr>
<tr>
<td>Median</td>
<td>1.0628e+07</td>
<td>2.2491e+06</td>
</tr>
<tr>
<td>Worst</td>
<td>2.5181e+07</td>
<td>4.8510e+06</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>5.5476e+06</td>
<td>7.6490e+05</td>
</tr>
<tr>
<td>Average error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum</td>
<td>9.3035e–05</td>
<td>1.3687e–08</td>
</tr>
<tr>
<td>Median</td>
<td>1.2113e–03</td>
<td>1.3687e–08</td>
</tr>
<tr>
<td>Maximum</td>
<td>6.5139e–02</td>
<td>8.1499e–03</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>1.64e–02</td>
<td>2.0000e–03</td>
</tr>
</tbody>
</table>

When a run is terminated, the accuracy of the obtained solutions to the desired optima becomes better than the limiting value used for deciding whether an optimum is found or not. With the new parameters, the obtained accuracy (mean error in objective value from exact optimal objective values) is better.
6 Constrained Multimodal Optimization

Most studies on evolutionary multimodal optimization concentrated on solving unconstrained multimodal problems, and some unconstrained test problem generators (Rönkkönen et al., 2008) now exist. To our knowledge, a systematic study on suggesting test problems on constrained multimodal optimization problems and algorithms to find multiple optima which lie on constraint boundaries does not exist. However, practical optimization problems most often involve constraints and in such cases multiple optima are likely to lie on one or more constraint boundaries. It is not clear (and has not yet been demonstrated well) whether existing evolutionary multimodal optimization algorithms are adequate to solve such constrained problems. In this paper, we first design scalable test problems for this purpose and then extend our proposed H-J-based evolutionary multimodal optimization algorithm to handle constraints. Simulation results are then presented on the test problems.

6.1 Constrained Multimodal Test Problems

Here, we suggest a simple test problem construction procedure which is extendable to an arbitrary dimension of the search space and has direct control on the number of global and local optima that the test problem will have. To illustrate the construction procedure, we first consider a two-variable problem. There are two nonconvex constraints and the minimum solutions lie at the intersection of the constraint boundaries. We call these problems CMMP($n$, $G$, $L$), where $n$ is the number of variables, $G$ is the number of global minima, and $L$ is the number of local minima in the problem.

PROBLEM CMMP(2,4,0):

\[
\begin{align*}
\text{minimize} & \quad f(x_1, x_2) = x_1^2 + x_2^2, \\
\text{subject to} & \quad g_1(x_1, x_2) = 4x_1^2 + x_2^2 \geq 4, \\
& \quad g_2(x_1, x_2) = x_1^2 + 4x_2^2 \geq 4, \\
& \quad -3 \leq x_1 \leq 3, \quad -3 \leq x_2 \leq 3.
\end{align*}
\]  

Figure 22 shows the constraint boundaries and the feasible search space (shaded region). Since the square of the distance from the origin is minimized, there are clearly four global minima in this problem. These points are intersection points of both ellipses (denoting
Figure 23: Two global and two local minimum points for the CMMP(2,2,2) problem.

the constraint boundaries) and are shown in the figure with circles. The optimal points are at \( x_1^* = \pm \sqrt{0.8} \) and \( x_2^* = \pm \sqrt{0.8} \) with a function value equal to \( f^* = 1.6 \). The contour lines of the objective function show that all four minimum points have the same function value.

Interestingly, if we modify the objective function as \( f(x_1, x_2) = x_1^2 + (x_2 - 0.2)^2 \) (problem CMMP(2,2,2)), there are two global minima: \((\sqrt{0.8}, \sqrt{0.8}), (-\sqrt{0.8}, \sqrt{0.8})\) with a function value of \( f^*_G = 1.282 \) and there are two local minima: \((\sqrt{0.8}, -\sqrt{0.8}), (-\sqrt{0.8}, -\sqrt{0.8})\) with a function value of \( f^* = 1.998 \). Figure 23 shows the contour of the objective function starting with a zero value at \((0, 0, 0.2)\) and then increasing with a point’s distance from \((0, 0, 0.2)\). The global and local minima are also marked in the figure.

If the objective function is modified as \( f(x_1, x_2) = (x_1 - 0.3)^2 + (x_2 - 0.2)^2 \) (the problem CMMP(2,1,3)), there are three different local minima and one global minimum, as shown in Figure 24. The global minimum is at \((\sqrt{0.8}, \sqrt{0.8})\) with a function value of \( f^*_G = 0.836 \). The three local minima are \((\sqrt{0.8}, -\sqrt{0.8}), (-\sqrt{0.8}, -\sqrt{0.8})\) with a function value of \( f^*_L = 1.551 \), \((-\sqrt{0.8}, \sqrt{0.8})\) with a function value of \( f^* = 1.909 \), and \((-\sqrt{0.8}, -\sqrt{0.8})\) with a function value of \( f^* = 2.624 \).

6.2 Higher Dimensional Constrained Test Problems

For higher dimensional (with \( n \) variables) search spaces, the above concept can be extended with higher dimensional ellipses as constraint boundaries and with more constraints. The following construction can have \( J = 1 \) to at most \( J = n \) constraints. For \( J \) constraints, there are \( 2^J \) minima. In the following problem, we have \( 2^J \) global minima:

\[
\text{minimize} \quad f(x) = \sum_{i=1}^{n} x_i^2, \\
\text{subject to} \quad g_j(x) = x_1^2 + 4x_2^2 + 9x_3^2 + \cdots + n^2x_n^2 \geq n^2, \\
\text{CMMP}(n, 2^J, 0); \quad g_2(x) = n^2x_1^2 + x_2^2 + 9x_3^2 + \cdots + (n - 1)^2x_n^2 \geq n^2, \\
\vdots \quad g_J(x) = C_{J,1}x_1^2 + C_{J,2}x_2^2 + C_{J,3}x_3^2 + \cdots + C_{J,n}x_n^2 \geq n^2, \\
-(n + 1) \leq x_i \leq (n + 1), \quad \text{for } i = 1, 2, \ldots, n.
\]
where
\[
C_{j,k} = \begin{cases} 
(n - j + k + 1) \mod n, & \text{if } (n - j + k + 1) \mod n \neq 0, \\
n, & \text{otherwise}.
\end{cases}
\]

In order to find the minimum points, we realize that they will lie on the intersection of constraint boundaries. Thus, we construct the Lagrange function:

\[
L(x, v) = f(x) - \sum_{j=1}^{J} v_j (g_j(x) - n^2),
\]

and set its partial derivatives to zero: \( \partial L / \partial x_i = 0 \) and \( \partial L / \partial v_j = 0 \). The second set of conditions results in the constraint conditions themselves. The first set of \( n \) equations is as follows:

\[
x_i^* \left( 1 + \sum_{j=1}^{J} v_j C_{j,i}^2 \right) = 0, \quad \text{for } i = 1, \ldots, n.
\]

For \( J < n \), \( n - J \) variables \( (x_i) \) will take a zero value to satisfy the above equations. From the remaining \( J \) linear equations, we can compute \( J \) optimal Lagrange multipliers \( v_j \). To find the remaining \( J \) optimal variable values \( x_i^* \), we set \( x_j = 0 \) for \( (n - J) \) chosen variables in constraint conditions. This process will lead to \( J \) equations with \( J \) variables \( (x_i) \), which can be solved to find \( x_i^* \) values. It turns out that by choosing different combinations of \( (n - J) \) variables to take a zero value, there will be multiple solution or solutions to the above set of equations. We then compute the function values of each solution and choose the solution or solutions which make the function value the smallest. It turns out that the smallest objective value occurs for \( x_i^* = 0, i = J, J + 1, \ldots, (n - 1) \). The remaining \( x_i^* \) variables for \( i = \{1, \ldots, (J - 1), n\} \) take nonzero values. We have computed these values for \( J = 1 \) to \( J = 4 \) and present the results below.

1. For one constraint \( (J = 1) \) leading to the problem \( \text{CMMP}(n, 2, 0) \), there are two global minima: \((0, 0, \ldots, 1)\) and \((0, 0, \ldots, -1)\) having a function value of \( f^* = 1 \).
For two constraints \((J = 2)\) leading to \(\text{CMMP}(n, 4, 0)\), there are four global minima:

\[
x_1^* = \pm \sqrt{\frac{n^2(2n - 1)}{(n^2 + n - 1)(n^2 - n + 1)}},
\]
\[
x_i^* = 0, \quad \text{for } i = 2, \ldots, (n - 1),
\]
\[
x_n^* = \pm \sqrt{\frac{n^2(n^2 - 1)}{(n^2 + n - 1)(n^2 - n + 1)}},
\]
\[
f^* = \frac{n^2(n^2 + 2n - 2)}{(n^2 + n - 1)(n^2 - n + 1)}.
\]

For example, for the \(n = 3\) variable \(J = 2\) constraint \(\text{CMMP}(3, 4, 0)\) problem, \(x_1^* = \pm \sqrt{45/77}\), \(x_2^* = 0\), and \(x_3^* = \pm \sqrt{72/77}\). Each of the four minimum points has a function value of 117/77 or 1.519. Two constraints and the corresponding four minimum points are shown in Figure 25 for the three-variable problem. As the number of variables \((n)\) increases, \(x_1^*\) approaches zero, \(x_n^*\) approaches one, and the optimal function value approaches one.

For three constraints \((J = 3)\) leading to \(\text{CMMP}(n, 8, 0)\), there are eight global minima:

\[
x_1^* = \pm \sqrt{\frac{n^2(2n^3 - n^2 + 4n - 8)}{n^6 - 2n^4 + 4n^3 + 7n^2 - 20n + 8}},
\]
\[
x_2^* = \pm \sqrt{\frac{n^2(2n^3 + n^2 - 6n + 4)}{n^6 - 2n^4 + 4n^3 + 7n^2 - 20n + 8}},
\]
\[
x_i^* = 0, \quad \text{for } i = 3, \ldots, (n - 1),
\]
\[
x_n^* = \pm \sqrt{\frac{n^2(n^4 - 2n^2 - 6n + 4)}{n^6 - 2n^4 + 4n^3 + 7n^2 - 20n + 8}},
\]
\[
f^* = \frac{n^3(n^3 + 4n^2 - 2n - 8)}{n^6 - 2n^4 + 4n^3 + 7n^2 - 20n + 8}.
\]
Figure 26: Eight global minimum points of the three-variable, three-constraints CMMP(3,8,0) problem.

Figure 26 shows three constraints and location of eight minimum points for the three-variable problem.

4. For $J = 4$ (CMMP($n$, 16, 0)), there are 16 global minima:

$$x_1^* = \pm \sqrt{\frac{n^2(2n^4 - n^3 + 12n^2 - 24n - 24)}{n^2 - 3n^5 + 6n^4 + 34n^3 - 80n^2 - 60n + 96}},$$

$$x_2^* = \pm \sqrt{\frac{n^2(2n^4 + n^3 - 2n^2 - 24)}{n^2 - 3n^5 + 6n^4 + 34n^3 - 80n^2 - 60n + 96}},$$

$$x_3^* = \pm \sqrt{\frac{n^2(2n^4 + 3n^3 - 12n^2 - 4n + 24)}{n^2 - 3n^5 + 6n^4 + 34n^3 - 80n^2 - 60n + 96}},$$

$$x_i^* = 0, \quad \text{for } i = 4, \ldots, (n - 1),$$

$$x_n^* = \pm \sqrt{\frac{n^2(5n^5 - 3n^3 - 22n^2 + 4n + 24)}{n^2 - 3n^5 + 6n^4 + 34n^3 - 80n^2 - 60n + 96}}.$$

For a 10-variable, four-constraint problem (CMMP(10, 16, 0)), the minimum points are ($\pm 0.4514$, $\pm 0.4608$, $\pm 0.4718$, 0, 0, 0, 0, 0, 0, $\pm 0.9846$) having a function value of 1.6081.

5. Similarly, the minima for other $J (\geq 5)$ values can also be computed. Finally, for $n$ constraints ($J = n$) leading to the CMMP($n$, $2^n$, 0) problem, there are $2^n$ global minima:

$$x_i^* = \pm \sqrt{\frac{6n}{(n + 1)(2n + 1)}}, \quad \text{for all } i = 1, 2, \ldots, n,$$

$$f^* = \frac{6n^2}{(n + 1)(2n + 1)}.$$

For example, for a five-variable, five-constraint problem (CMMP(5, 32, 0)), $x_i^* = \pm 0.6742$ for $i = 1, \ldots, 5$, with a function value of 2.2727.
Table 5: Multimodal points found by constraint handling procedure for different problems.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution 1</th>
<th>Solution 2</th>
<th>Solution 3</th>
<th>Solution 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMMP(2,4,0)</td>
<td>x</td>
<td>(0.894, 0.895)</td>
<td>(–0.895, 0.895)</td>
<td>(0.894, –0.896)</td>
</tr>
<tr>
<td></td>
<td>f</td>
<td>1.601</td>
<td>1.601</td>
<td>1.602</td>
</tr>
<tr>
<td>CMMP(2,2,2)</td>
<td>x</td>
<td>(0.894, 0.895)</td>
<td>(0.895, –0.894)</td>
<td>(0.894, –0.895)</td>
</tr>
<tr>
<td></td>
<td>f</td>
<td>1.583</td>
<td>1.584</td>
<td>1.619</td>
</tr>
<tr>
<td>CMMP(2,1,3)</td>
<td>x</td>
<td>(0.896, 0.894)</td>
<td>(–0.897, 0.894)</td>
<td>(0.894, –0.895)</td>
</tr>
<tr>
<td></td>
<td>f</td>
<td>1.548</td>
<td>1.587</td>
<td>1.619</td>
</tr>
</tbody>
</table>

As in the case of the two-variable problem described in the previous subsection, the objective function in a higher dimensional problem can also be modified by shifting the zero of the function away from the origin to introduce different types of local minima.

6.3 Constrained Handling Procedure for Multimodal Optimization

To take care of feasible and infeasible solutions present in a population, we use the constraint handling strategy of the NSGA-II procedure (Deb, 2001; Deb et al., 2002). It is worth mentioning here that other multi-objective constraint handling procedures (Coello and Montes, 2002; Deb, 2001) can also be used here. When two solutions are compared for domination, one of the following three scenarios can happen: (i) when one solution is feasible and other is infeasible, we choose the feasible solution, (ii) when two infeasible solutions are being compared, we simply choose the solution having smaller overall normalized constraint violation, and (iii) when both solutions are feasible, we follow the niching comparison method used before for unconstrained problems. Instead of using a procedure which is completely different from the unconstrained method, we use a simple modification to our proposed approach and importantly without the need for any additional constrained handling parameter. To get an accurate evaluation of \( f_2 \) for solutions close to a constraint boundary, we ignore any infeasible solution in the count of better neighboring solutions. The H-J exploratory search is used for evaluating the second objective function.

6.4 Constraint Handling Results

In this subsection, we construct a number of constrained multimodal optimization problems using the above procedure and attempt to solve them using the proposed procedure. Identical NSGA-II parameter values as those used in the unconstrained cases are used here, except that we increase the population size to \( N = 25 \max(n, M) \), due to a better handling of nonlinear and nonconvex constraints here. For all constrained problems, we use \( \delta_f = 0.2 \) and \( \delta_x = 0.08 \). An optimum is considered to be found if a feasible population member within a Euclidean distance of 0.05 from the true location of the optimum is obtained. In all cases, we perform 30 runs, but due to brevity, we present the obtained multiple solution vectors from a single simulation run.

6.4.1 Problem CMMP(2,4,0)

Table 5 presents the four minimum points found by the proposed procedure. By comparing with Figure 22, it can be observed that the obtained points are identical to the four minimum points.

6.4.2 Problem CMMP(5,4,0)

Figure 27 presents four minimum points found by the proposed procedure for the five-variable, two-constraint CMMP(5,4,0) problem. This figure allows an interesting way
Figure 27: Four global minimum points for the five-variable, two-constraint CMMP(5,4,0) problem.

Figure 28: Thirty-two global minimum points for the five-variable, five-constraint CMMP(5,32,0) problem.

to visualize multiple solutions for more than two variables. The $x$-axis indicates the variable number (1 to 5, in this case) and the $y$-axis denotes the value of the variable. Thus, a polyline joining the variable values at different variable IDs represents one of the solutions found by the proposed procedure. It is interesting to note that the three intermediate variables ($x_2$ to $x_4$) take a zero value for all four solutions. Four solutions are formed from two values each from $x_1$ and $x_5$. The four solutions have the following objective values: 1.355, 1.356, 1.355, and 1.356. Although theoretically all these values should be identical, the obtained solutions differ only in their third decimal places in their function values.

6.4.3 Problem CMMP(5,32,0)
Next, we consider the five-variable, five-constraint problem, which has 32 minimum points having an identical objective value. Figure 28 represents all 32 solutions arising
K. Deb and A. Saha

6.4.4 Problem CMMP(5,1,31)

Next, we include a problem having a combination of global and local constrained minima. The objective function in CMMP(5,32,0) is modified to construct CMMP(5,1,31), as follows:

$$f(x) = \sum_{i=1}^{5} (x_i - 0.05i)^2.$$  \hspace{1cm} (12)

In this problem, we have only one global minimum and 31 other local minima, all lying on intersection on constraint boundaries. Figure 29 plots the function value of the obtained minimum points and the corresponding theoretical minimum function values. It is clear that out of the 32 minimum points, the proposed procedure is able to find the best 31 of them (based on objective function value) and misses the local minimum point having the worst objective value. The figure represents a typical run obtained with 10 different initial populations. This study shows that despite the presence of multiple constrained local minimum points, the proposed procedure is able to locate all but one of the minima.

6.4.5 Problem CMMP(10,16,0)

Figure 30 shows the obtained solutions on the 10-variable, four-constraint CMMP(10,16,0) problem. The intermediate six variables ($x_4$ to $x_9$) take a value close to zero and 16 solutions come from each of two values of the other four variables. The objective function values among the 16 solutions vary only a maximum of 0.43% in the range [1.614, 1.621].

7 Conclusions

In this paper, we have suggested a bi-objective formulation of a multimodal optimization problem so that multiple global and local optimal solutions become the sole candidates for the weak Pareto-optimal set. While the objective function of the multimodal optimization problem is one of the objectives, a couple of suggestions of the
Multimodal Optimization Using a Bi-Objective Evolutionary Algorithm

Second objective function have been made here. Starting with the gradient-based approaches (demonstrating the principle of the proposed bi-objective approach), more pragmatic neighborhood-count-based approaches have been systematically developed for this purpose. Modifications to an existing EMO procedure were made in order to find weak nondominated solutions. Using the H-J pattern search-based method for the second objective, the proposed EMO procedure has been able to solve as large as 16-variable, 48-optima problems having a combination of global and local optima.

Another hallmark of this study is the suggestion of a multimodal constrained test problem generator which is scalable in terms of the number of variables, the number of constraints, and the number of optima. The unconstrained multimodal optimization procedure developed in this study has been adapted to solve 10-variable, 16-optima constrained problems with success. The existing multimodal optimization literature did not include a systematic study of constrained problems. The construction procedure of the test problem generator and the development of the proposed algorithm should motivate researchers to pursue constrained multimodal studies further.

The proposed study can be extended in a number of ways. The H-J neighborhood count procedure requires \(2^n\) evaluations around each solution. But the best neighboring point around a solution can be used to replace the current solution, thus performing a repair operation, which may speed up convergence to the optima. Further reduction in computational complexity may be possible by stopping the H-J exploratory search when it fails to find an adequate number of better neighboring solutions within a fraction of the total \(2^n\) allocations. Both of these ideas are worth trying out and if successful, the number of function evaluations can be decreased considerably.

This study explores a few forms of an additional objective, which in conjunction with the original objective function of the multimodal problem, uniquely maps the optimal solutions on the weak Pareto-optimal front. The idea is new and interesting and other more efficient forms of the additional objective function can now be designed and explored.

Figure 30: Sixteen global minimum points for the 10-variable, four-constraint CMMP (10,16,0) problem.
Acknowledgments

The authors wish to thank Prof. Juergen Branke for sharing some initial thoughts with the first author during his Bessel research award visit to the University of Karlsruhe in 2003. This study is funded by the Department of Science and Technology, Government of India under SERC-Engineering Sciences scheme (No. SR/S3/MERC/091/2009).

References


