Struct Multidisc Optim 24, 351–361 © Springer-Verlag 2002
Digital Object Identifier (DOI) 10.1007/s00158-002-0247-6

Entropy-based multi-objective genetic algorithm for design optimization

A. Farhang-Mehr and S. Azarm

Abstract Obtaining a fullest possible representation of solutions to a multiobjective optimization problem has been a major concern in Multi-Objective Genetic Algorithms (MOGAs). This is because a MOGA, due to its very nature, can only produce a discrete representation of Pareto solutions to a multiobjective optimization problem that usually tend to group into clusters. This paper presents a new MOGA, one that aims at obtaining the Pareto solutions with maximum possible coverage and uniformity along the Pareto frontier. The new method, called an Entropy-based MOGA (or E-MOGA), is based on an application of concepts from the statistical theory of gases to a baseline MOGA. Two demonstration examples, the design of a two-bar truss and a speed reducer, are used to demonstrate the effectiveness of E-MOGA in comparison to the baseline MOGA.

Key words multiobjective optimization, genetic algorithms, entropy

1 Introduction

Several recent techniques have been developed to improve the coverage and uniformity of solutions obtained by Multi-Objective Genetic Algorithms (MOGAs) (e.g. Camponogara and Talukdar 1997; Reynolds 2000). In particular, these algorithms attempt to generate additional solutions to fill in under-represented areas along the Pareto optimal frontier. The basic concept in these techniques is to randomly select and project some candidate points near the edge of a gap to fill in the interior or exterior voids in a solution set. Unfortunately, it may be difficult to judge whether or not such heuristics will in general improve the overall solution quality along the Pareto frontier. However, one can find an analogy in the statistical theory of gases, which can be taken advantage of to improve the coverage and uniformity of the solutions obtained by a MOGA.

When an ideal gas undergoes an expansion inside of an enclosure, the gas molecules move randomly (i.e. without having any a priori information about the geometry of the enclosure) and achieve a homogenous and uniform equilibrium state with maximum entropy (Fay 1965). One may apply this analogy of the expansion of an ideal gas in an enclosure to the evolution of solutions obtained by a MOGA in order to achieve uniformly distributed solutions with a maximum possible coverage along the Pareto frontier.

Some other similarities can also be observed between a Genetic Algorithms (GA) and Statistical Thermodynamics (ST). Both GA and ST describe statistical evolution of a population (i.e. individual designs in the GA and molecules in the ST), define the microscopic and macroscopic specifications of a system, and introduce constraints. Indeed, statistical thermodynamics provides a favourable platform for emulating the genetic algorithms. A formalism recently introduced by Prugel-Bennett and Shapiro (1994, 1997) uses the methods of statistical thermodynamics to analyze the dynamics of GA operators for maximum entropy in a single objective GA (also Shapiro *et al.* 1994, 1995). Moreover, Kita *et al.* (1996) as well as Cui *et al.* (2001) took advantage of the thermodynamical notion of energy and entropy to prevent premature convergence of the population to the Pareto optimal frontier and preserve diversity. Based on these concepts, a fitness scaling strategy is introduced (Kita *et al.* 1996) to control the genetic similarity and diversity of the individuals by postponing the maturity of the population. However, postponing the maturity may not always be desirable since it increases the execution time of a MOGA. As such, Sobieski *et al.* (1998) suggested an alternative operator, instead of regular crossover and mutation operators, that was based on projection of solution points according to a bell-curve distribution. The use of this alternative operator could provide better diversity in the population without degrading the convergence rate of the optimization process. However, no research has been reported in the literature that applies the no-

Received June 29, 2001

A. Farhang-Mehr and S. Azarm

Department of Mechanical Engineering, University of Maryland, College Park, MD 20742-3035, USA
e-mail: azarm@eng.umd.edu

tion of statistical thermodynamics to create enhanced operators (see, for example, Fonseca and Fleming (1995) and Coello Coello (1999) for a comprehensive review of the evolutionary based multiobjective optimization approaches). Such an operator (along with regular crossover and mutation operators) can articulate the diversification of individuals in the population such that a state of maximum entropy (i.e. maximum possible diversity in the population) is achieved. The rest of the paper is devoted to this issue. The new MOGA emulates the distribution of solutions along a Pareto frontier with the expansion of an ideal gas in an enclosure to reach a uniformly-distributed state with maximum entropy.

2 Entropy-based multiobjective genetic algorithm (E-MOGA)

In this section, we present an Entropy-based MOGA (or E-MOGA). We briefly review a statistical model for an ideal gas and then devise the E-MOGA according to this model. This is followed by a detailed description of the E-MOGA.

2.1 Statistical model of an ideal gas

As a background (Toda *et al.* 1991), consider an ideal gas that consists of a set of moving molecules. Suppose that the vectors $\mathbf{x}$ and $\mathbf{v}$ give the position and velocity of a molecule in the gas, respectively. The state-space is defined by a $2n$-dimensional domain whose coordinates are $x_1, \ldots, x_n$, and $v_1, \ldots, v_n$, where n is the dimension of the space. (For example, in the real world, n is equal to 3.) A point in this domain represents the state of each molecule at some instant, and the state of the entire collection of molecules is represented by a set of points. One particular arrangement of these points in the state-space is called a microstate. Each microstate has an external property (such as total energy) that is called a macrostate. Hence, each microstate corresponds to a particular macrostate, but there are many possible microstates that correspond to the same macrostate (Desloge 1996; Penrose 1970).

It is assumed that an ideal gas in its equilibrium state is homogeneous, isotropic and time-invariant. If one adds an additional assumption that all microstates whose corresponding macrostate satisfies some imposed boundary conditions (i.e. the constraints) have the same probability to occur, then at the equilibrium state, the Maxwellian distribution is obtained (Fay 1965):

$$f(\mathbf{x}, \mathbf{v}, t) = n \left(\frac{\lambda}{\pi}\right)^{\frac{3}{2}} \exp\left(-\lambda v^2\right) , \tag{1}$$

where v is the magnitude of the velocity vector, $\mathbf{v}$, and the distribution function, $f(\mathbf{x}, \mathbf{v}, t)$, is defined as the density of points in the state-space. (Note that this is different from the density of molecules in the position-space.) The quantity λ is a positive constant. However, if one is interested in the number of molecules with a speed between v and $v + dv$, then f can be integrated to obtain the velocity distribution, f_v. For example in a three-dimensional enclosure, the velocity distribution function is

$$f_v(\mathbf{x}, \mathbf{v}, t) = 4\pi v^2 f(\mathbf{x}, \mathbf{v}, t) , \tag{2}$$

which indicates the density of the molecules in the velocity-space (whose coordinates are the velocity components of the molecules). This distribution is used later on in this paper to assign a velocity to the individuals in the E-MOGA.

One can define the thermodynamic probability, W, of a macrostate as the number of microstates associated with it. The entropy, H, of this state is then defined as

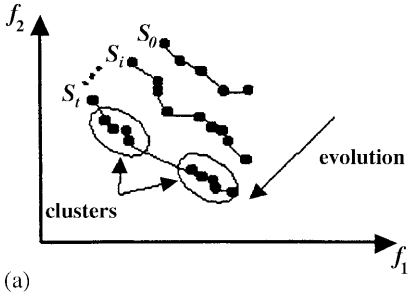
$$H = k \ln(W) , \tag{3}$$

where k is a constant. One can think of the entropy as a system attribute that indicates how the system will thermodynamically evolve (Desloge 1996). Then for each state of the gas, one can define a single-valued function with the following property: if a constraint is imposed on the system or removed from it, the system will evolve to its new equilibrium state with a maximum entropy. This concept of entropy is used to develop an E-MOGA, as described in the following.

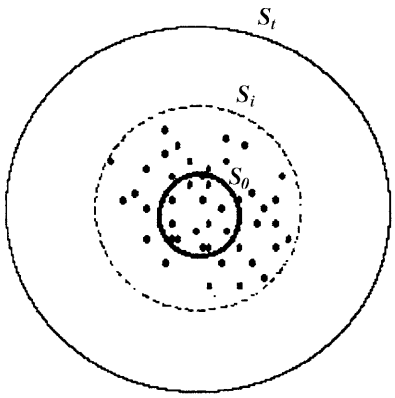
2.2 Development of E-MOGA via analogy with the ideal gas model

Consider the operations of a two-objective MOGA, as shown in Fig. 1a. We designate the set of non-dominated points after t iterations as S_t. (Nondominated points for a given population refer to the best possible points in the population among which there exist tradeoffs, i.e. from one point another along the nondominated points, an improvement in the value of one objective translates into the degradation of the value of the other objective.) As the population evolves from the nondominated points for the initial population, S_0, to the nondominated points after t iterations, S_t, the Pareto solution set or frontier is formed. The "iteration number" in the E-MOGA, t, corresponds to "time" in statistical thermodynamics. As shown in Fig. 1a, the Pareto solution set may consist of several clusters of points and may not cover the entire range of Pareto frontier. Clustering is a well-known phenomenon in a MOGA wherein solutions that are desirable to be spread evenly along the Pareto frontier are instead grouped together.

As can be seen in Fig. 1b, a direct analogy can be observed between the operation of a MOGA and an ideal gas undergoing an expansion in an enclosure. According to the definition of an ideal gas, there is no interaction between the molecules of an ideal gas in an enclosure.



(a)



(b)

Fig. 1 Analogy of: (a) evolution of the solutions in a MOGA (b) spread of an ideal gas molecules in an enclosure

That is, the molecules move along random directions until they collide with the boundary of the enclosure. As a result of the collision with the boundary, the molecules are reflected back into the enclosure with the same velocity they had before the collision but along a different direction following the "mirror's law". As we mentioned before, if the velocity distribution of such a system follows the Maxwellian distribution, the set of particles will automatically evolve to maximize the entropy and consequently result in the uniformity of coverage of the molecules in the enclosure. To an outside observer, the gas expands gradually until it fills the enclosure and eventually reaches a uniform, homogenous and time-invariant state. Hence, if we can simulate this behavior of the ideal gas and apply it to a MOGA to control the expansion of points along the Pareto frontier, then the final state of solutions along the Pareto frontier will be as much uniform as possible, producing the fullest possible representation of the Pareto set, instead of a set of clustered points.

The objective of this paper is therefore to modify a MOGA so that it enables an expansion of solutions in lateral directions. (A lateral direction is defined in Sect. 2.2.2. In short, it is a random direction normal to the evolution direction shown in Fig. 1(a).) At the same time the evolution of solutions takes place with the GA operations. To achieve the objective of this paper, we proceed with some modifications to a baseline MOGA, as described in the following sections. [The baseline MOGA, hereafter called MOGA-NA, was recently developed by Narayanan and Azarm (1999)].

2.2.1 Velocity assignment

A velocity value is assigned to every individual of the population in an E-MOGA, according to the ideal gas model. At the beginning of the process, each individual is labelled with a velocity according to the Maxwellian velocity distribution in (2). This velocity will remain constant during the entire optimization process. This is similar to the case of the ideal gas, where the magnitude of velocity of a molecule does not change but its direction changes as it collides with the boundaries of the enclosure.

2.2.2 Transverse expansion hyper-surface

MOGA operations mainly consist of evolving the population via the GA operators of crossover and mutation, as it approaches the Pareto frontier. To extend the MOGA operations so that it also simulates an expanding ideal gas in an enclosure, at the beginning of each iteration the population is expanded laterally, normal to the progress vectors. Progress vectors are defined as the vectors along which the individuals in the population are expected to evolve. The progress vector is different for each point in the population and should be estimated individually. To estimate this vector, we propose an algorithm that is based on the relative position and fitness of points in the objective space while all operations are performed in the variable space.

Consider the two-objective optimization problem shown in Fig. 2, wherein there are N_1 non-dominated solution points in the population at its current stage of evolution. We label these points as $P_1, P_2, \ldots, P_{N_1}$. If we eliminate these points from the population, we obtain N_2 nondominated points, denote these as P_{N_1+1}, $P_{N_1+2}, \ldots,$ $P_{N_1+N_2}$. As we repeat this process, the population is partitioned into several sets of points. We assign a ranking of one to the first set (i.e. N_1 points, the fittest points in the current population), a ranking of 2 to the second set of points, and so on. Each set will form a curve or

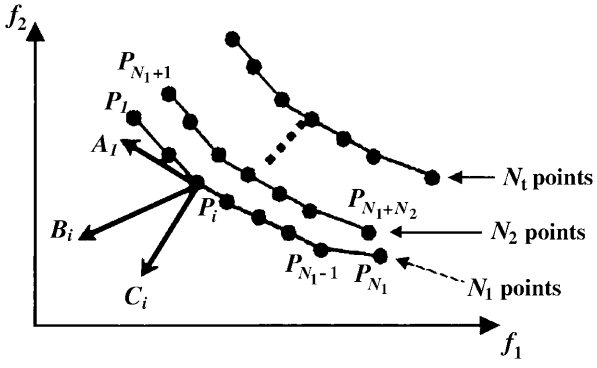


Fig. 2 Estimation of the progress vector in a two-dimensional objective-space

a hyper-surface for the higher dimensions of the objective space, hereafter referred to as a transverse-expansion hyper-surface.

The lateral expansion of the solutions is done in a direction tangent to this hyper-surface. As the solutions are evolved, these transverse hyper-surfaces gradually converge to the Pareto frontier. (These hyper-surfaces will be (m-1)-dimensional in an m-dimensional objective space. For example as shown in Fig. 3, the transverse hyper-surface of a three-objective problem will be a two-dimensional surface on which the population is expanded laterally.)

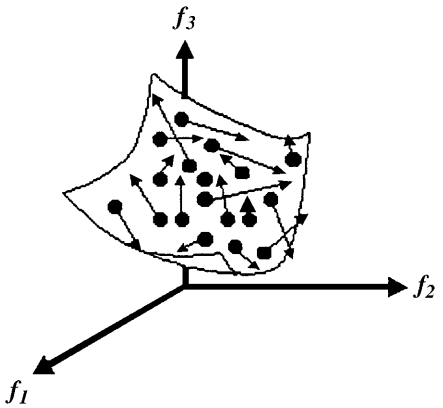


Fig. 3 Transverse expansion of individuals in E-MOGA

Consider the point P_i on the first nondominated set ($1 \le i \le N_1$), see Fig. 2. We denote the vector connecting P_k to P_i as $\mathbf{Z}_{ki}$. We define an offset vector $\mathbf{A}_i$ as

$$\mathbf{A}_i = \frac{\sum_{k=1}^{N_1} \mathbf{Z}_{ki}}{N_1} \qquad (1 \le i \le N_1) \tag{4}$$

wherein $\mathbf{A}_i$ is an offset vector, since it is almost tangent to the transverse hyper-surface and points to the nearest edge of the nondominated set. In addition, the magnitude of this vector is larger for a point near the edge of the set while in the middle of the set its magnitude is small. Similarly, the vector $\mathbf{B}_i$ is defined

$$\mathbf{B}_i = \frac{\sum_{k=N_1+1}^{N_1+N_2} \mathbf{Z}_{ki}}{N_2} \qquad (1 \le i \le N_1)\,. \tag{5}$$

As can be seen in Fig. 2, since the points in N_1 are evolved (in terms of all objectives) as compared to the points in N_2, $\mathbf{B}_i$ consists of a normal-progress component in addition to the offset vector. If the points in the first and second non-dominated sets, N_1 and N_2 points in Fig. 2, respectively, are distributed in the same fashion, then the offset vectors due to both of these first and second nondominated sets are approximately equal. Hence, one can subtract the offset vector, $\mathbf{A}_i$ from $\mathbf{B}_i$ to obtain a progress vector

$$\mathbf{C}_i = \mathbf{B}_i - \mathbf{A}_i = \frac{\sum_{k=N_1+1}^{N_1+N_2} \mathbf{Z}_{ki}}{N_2} - \frac{\sum_{k=1}^{N_1} \mathbf{Z}_{ki}}{N_1} \quad (1 \le i \le N_1)\,, \tag{6}$$

where $\mathbf{C}_i$ estimates the progress vector of the point P_i (Fig. 2). So far, we have assumed that P_i is in the first non-dominated set, i.e. the N_1 points in the population. For a point P_i, in general, in the m-th nondominated set, we have

$$\mathbf{C}_i = \frac{\sum_{k=N_1+N_2+\ldots+N_m+1}^{N_1+N_2+\ldots+N_m+N_{m+1}} \mathbf{Z}_{ki}}{N_{m+1}} - \frac{\sum_{k=N_1+N_2+\ldots+N_{m-1}+1}^{N_1+N_2+\ldots+N_m} \mathbf{Z}_{ki}}{N_m}$$

where

$$(N_1 + \ldots + N_{m-1} + 1) \le i \le (N_1 + \ldots + N_m)\,. \tag{7}$$

It is clear that the above-mentioned estimate becomes less accurate for very inferior or poor ranking points and their corresponding hyper-surfaces since they may not consist of enough points to give an acceptable estimate of offset and progress vectors. However, as mentioned later in this paper, we are mainly interested in the good-ranking hyper-surfaces with better fitness and higher chance of reproduction.

Now that we have obtained $\mathbf{C}_i$, a lateral movement vector is chosen randomly in a direction normal to the progress vector of each individual, as shown in Fig. 4. Each point moves along its lateral movement vector until it violates a constraint (just like a molecule in an enclosure that moves along a straight line until it collides with the enclosure boundaries). This will be described later, in the constraint handling section.

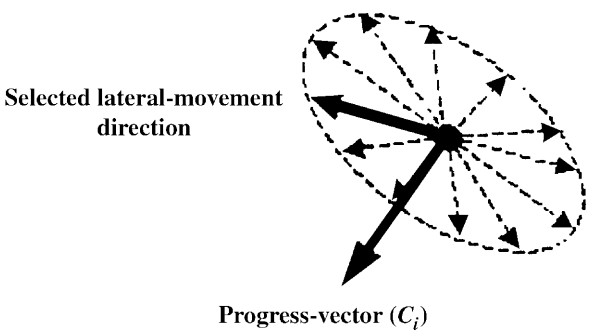


Fig. 4 Lateral-movement vector is chosen randomly, normal to the progress vector

2.2.3 Expansion operator

The expansion operator is applied at each iteration of the E-MOGA. This operator enhances each individual (or a certain percentage of the population) as follows.

1. A new chromosome (child) is generated by moving the chosen individual (single-parent) along its assigned lateral-movement vector in the variable-space.
2. The magnitude of the movement is proportional to its assigned velocity (as described in Sect. 2.2.1).

Since this operator is applied every iteration, it simulates the gradual movement and expansion of gas molecules with different velocity magnitudes and directions. However, there are two parameters that should be set in the E-MOGA, the expansion percentage and expansion start, as described in the following.

Expansion percentage. To keep the number of function calls as low as possible, we apply the expansion operator only to a certain percentage of the fittest individuals in the population. As these points are expanded to the new regions, they act as seeds for the new offspring in those regions.

Expansion start. As described in Sect. 2.2.2, the estimation of the progress vector is based on the relative position and fitness of the points. Specifically as the number of the first and second ranked nondominated points (i.e. N_1 and N_2 points in Fig. 2) increases, the corresponding progress and lateral-movement vectors can be estimated more accurately. Hence, if we let the population evolve for a few iterations before we start applying the lateral expansion operator, there will be more points in these sets resulting in a more accurate estimation of the progress and lateral-movement vectors. In the examples discussed at the end of this paper, for instance, the expansion starts at the fifth generations. Thereafter, the expansion operator is applied to all subsequent generations.

2.3 Constraint handling

When a gas molecule collides with the boundary of an enclosure, it is reflected back into the enclosure without a change in its velocity magnitude. However, the movement direction will be different and follows the mirror's reflection law (Fig. 5a). Similarly, a constraint for an E-MOGA is a hypersurface in the variable space and can be treated as a boundary (or a wall). This is shown in Fig. 5b.

Assume that all individuals in the initial population are feasible. Every time that the population undergoes an expansion (i.e. an expansion operator is applied), there might be several points that attempt to violate one or more constraints to enter the infeasible region. We reflect these points back into the feasible region according to the mirror's reflection law and assign to them a new lateral-velocity vector (Fig. 5b). This constraint handling aspect of E-MOGA makes it a "feasible-direction" type search method. This is because if all individuals in the initial population satisfy the constraints, all subsequent generations will remain within the feasible domain. To ensure that the initial population is feasible, the E-MOGA checks for infeasible points before the first generation undergoes expansion and replaces, via a random number generator, all infeasible individuals with feasible ones. This is continued until all individuals in the initial population are feasible.

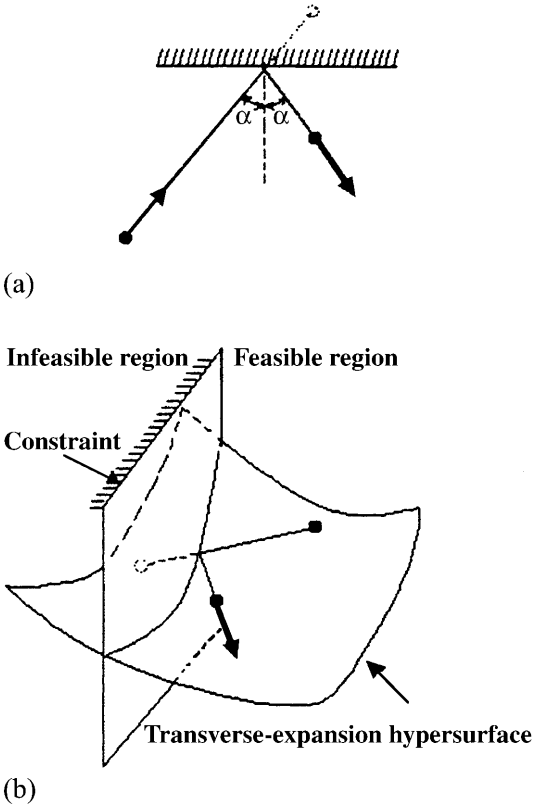


Fig. 5 (a) A molecule reflects back into the enclosure after it hits the enclosure's wall. (b) The solution point is reflected back to the feasible region if it violates the constraint

2.4 E-MOGA: step-by-step description

A detailed flowchart for the proposed E-MOGA is shown in Fig. 6. Below, the algorithm is described step-by-step.

Step 1. The initial population is generated randomly until all individuals are feasible.

Step 2. A velocity is assigned to each individual in the population according to the Maxwellian probability distribution function in (2). This velocity remains constant throughout the process. (See Sect. 2.2.1.)

Step 3. The individuals in the population are rank-ordered and the hyper-surfaces are created (i.e. sets N_1, N_2, ..., as described in Sect. 2.2.2)

Step 4. The progress vectors of the individuals are estimated from (7).

Step 5. A lateral movement vector is assigned to each individual randomly, normal to the progress vector obtained in Step 4.

Step 6. The expansion operator is applied to a prespecified percentage of the population. (See Sect. 2.2.3.)

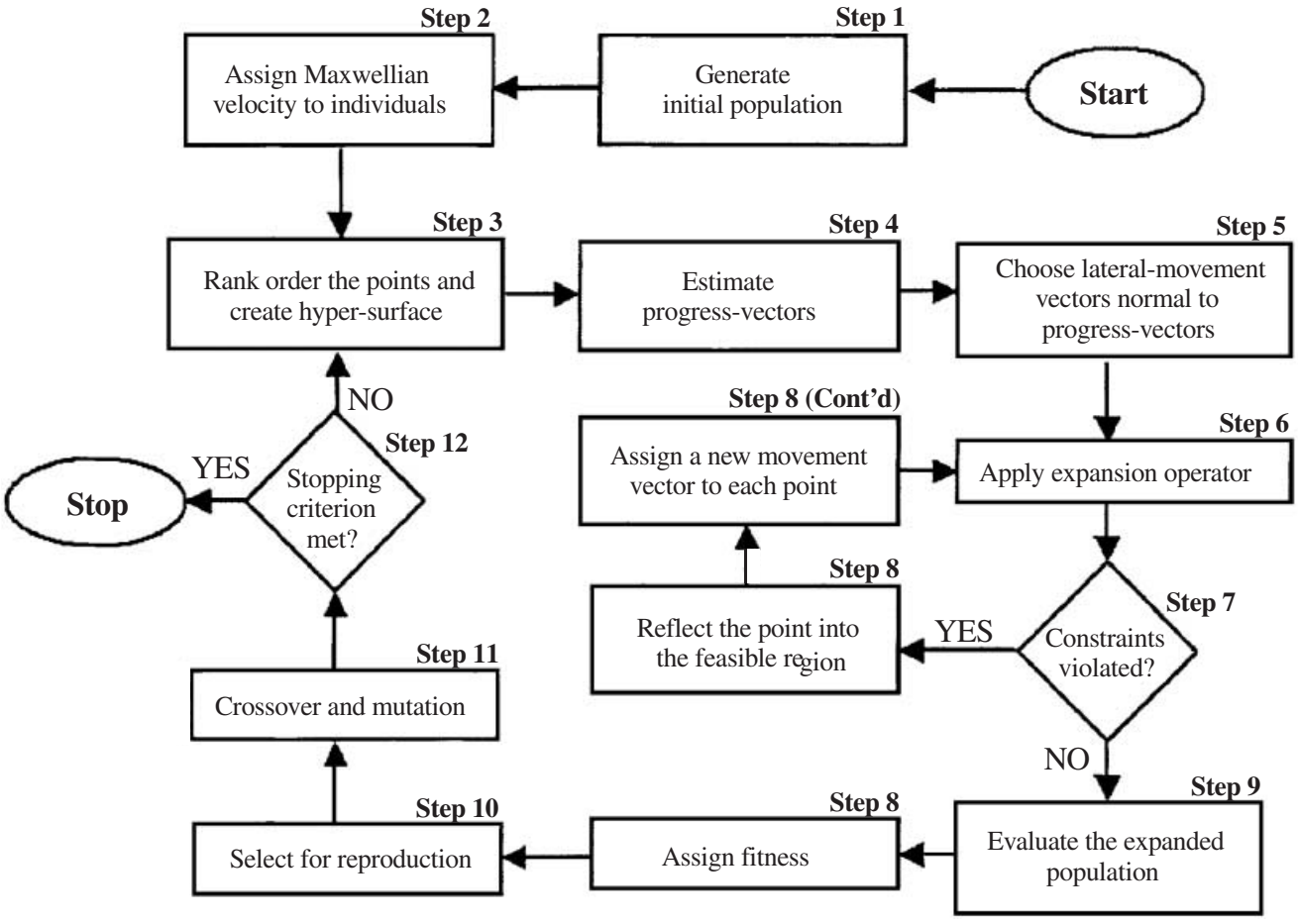


Fig. 6 Flowchart E-MOGA

Step 7. After the expansion is applied to the individuals, they are checked for constraint violations.

Step 8. If an individual violates a constraint, it is reflected back from the corresponding constraint boundary and a new movement direction is assigned according to the mirror reflection law (Sect. 2.3). Go back to Step 7.

Step 9. The expanded population is evaluated and a fitness value is assigned to each individual.

Step 10. A certain percentage of the individuals are selected for reproduction. The chance of being selected is higher for the fittest individuals.

Step 11. The selected parents undergo crossover and mutation. The offspring are added to the current population. Then the entire population is rank-ordered and the worst individuals are discarded.

Step 12. If the stopping criterion is not met, go to Step 3 and continue. Otherwise, stop.

As stated before for the ideal gas expansion, the population of chromosomes in E-MOGA is expected to gradually achieve the maximum-entropy macrostate that indicates a uniform probability distribution of microstates. During the process, as the individuals are expanded, the entropy of the population increases gradually. That is why we call this algorithm an "Entropy based MOGA" or E-MOGA.

3 Test examples

The proposed E-MOGA has been applied to two engineering examples, the design of a two-bar truss and a speed-reducer, for demonstration and comparison against a baseline MOGA.

3.1 Two-bar truss

The first example involves the design of a two-bar truss that was originally formulated as a single-objective problem by Kirsch (1981). The problem was reformulated later to demonstrate the application of a MOGA to multi-objective engineering problems (Narayanan and Azarm 1999). In this example, as illustrated in Fig. 7, the vertical position of point C and the cross-sectional areas of links AC and BC are to be selected and thus the design variables, which are all continuous, are x_1, x_2 and y. The design objectives are to minimize the total volume (and consequently the weight) of the structure and to minimize the tensile stress in link AC. The constraints are the maximum allowable stresses in AC and BC that should not exceed 100 000 kPa and the total volume of the ma-

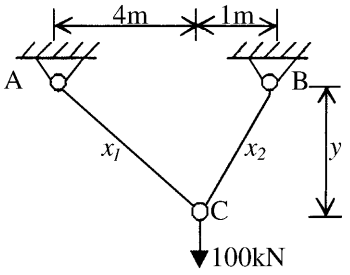


Fig. 7 Two-bar truss

terial used in the structure that should be held less than $0.1\,\mathrm{m}^3$.

Constraints are imposed on the design objectives (i.e. objective constraints), as shown in the formulation of the problem. This is because the Pareto set is asymptotic and extends from $-\infty$ to ∞. As x_1 and x_2 approach zero, f_{volume} goes to zero and $f_{\text{stress},AC}$ and $f_{\text{stress},BC}$ approach infinity. On the other hand, as x_1 and x_2 approach infinity, f_{volume} goes to infinity and $f_{\text{stress},AC}$ and $f_{\text{stress},BC}$ approach zero. Hence, in order to generate Pareto optimal solutions in a reasonable range, the objective constraints are imposed. The problem formulation is shown below

$$\text{minimize} \quad f_{\text{volume}} = x_1 \left(16+y^2\right)^{\frac{1}{2}} + x_2 \left(1+y^2\right)^{\frac{1}{2}} , \qquad (8)$$

$$\text{minimize} \quad f_{\text{stress},AC} = \frac{20\left(16+y^2\right)^{\frac{1}{2}}}{yx_1}$$

s.t.

$$f_{\text{volume}} \le 0.1$$

$$f_{\text{stress},AC} \le 100\,000$$

$$f_{\text{stress},BC} \le 100\,000$$

$$1 \le y \le 3$$

$$x_1, x_2 > 0$$

$$\text{where } f_{\text{stress},BC} = \frac{80\left(1+y^2\right)^{\frac{1}{2}}}{yx_2} .$$

The problem has been solved with both the baseline MOGA (MOGA-NA; Narayanan and Azarm 1999) and E-MOGA. The value of the GA parameters used to solve this problem is listed in Table 1.

Figure 8 illustrates the gradual evolution of solutions in both MOGA-NA (see Figs. 8a, c and e) and E-MOGA (see Figs. 8b, d and f). As shown in these figures, a significant improvement is observed in the results obtained by E-MOGA as compared with those by MOGA-NA. In the final Pareto-optimal set of MOGA-NA, Fig. 8e, the majority of solution points are clustered in two regions with the rest of the Pareto frontier is left empty or very sparingly populated. In contrast, Fig. 8f shows a more evenly distribution of solution points along the Pareto frontier in E-MOGA, without any noticeable clustering of the points. Moreover, comparing the range of the optimal solution set obtained from each technique, one can notice that E-MOGA covers a larger portion of the Pareto frontier. Hence, based on this example, it can be concluded that the Pareto set generated by E-MOGA is significantly better than that generated by MOGA-NA in terms of uniformity and coverage of the Pareto frontier.

Table 1 MOGA parameters in the two-bar truss example

Optimization technique	MOGA-NA	E-MOGA
population size	200	200
replacement per generation	10	10
function calls	550	550
crossover type	2-point	2-point
crossover probability	0.8	0.8
mutation probability	0.01	0.01
bits per variable	10	10
expansion percentage	N/A	10%
expansion start	N/A	5-th gen.
expansion finish	N/A	45-th gen.

3.2 Speed reducer

This example was originally formulated by Golinski (1970) as a single-objective optimization problem. The problem has been converted into a two-objective optimization problem [following Azarm *et al.* (1989) for a three-objective formulation]. The example represents the design of a simple gear-box, as shown in Fig. 9, that can be used in a light airplanes. The seven design variables in the formulation are: gear face width (x_1), teeth module (x_2), number of teeth of pinion (x_3 – integer variable), distance between bearings on shaft 1 (x_4), distance between bearings on shaft 2 (x_5), diameter of shaft 1 (x_6), and diameter of shaft 2 (x_7). The first design objective, f_1, is to minimize the volume. The second objective, f_2, is to minimize the stress in one of the two gear shafts.

The design is subject to a number of constraints imposed by gear and shaft design practices. An upper and lower limit is imposed on each of the seven design variables. There are 11 other inequality constraints (one of which is a constraint imposed on the first objective), as follows: g_1 is an upper bound of the bending stress of the gear tooth; g_2: upper bound of the contact stress of the gear tooth; g_3, g_4 are upper bounds of the transverse deflection of the shafts; g_5–g_7 are dimensional restrictions based on space and/or experience; g_8, g_9 are design requirements on the shaft based on experience; and g_{10}, g_{11} are constraints on stress in the gear shafts. The optimization formulation is

$$\text{minimize} \quad f_{\text{volume}} = f_1 = 7.854 x_1 x_2^2 \times$$

$$\left(\frac{x_3^2}{3} + 1.493 x_3 - 4.309\right) - 1.508 x_1 \left(x_6^2 + x_7^2\right) +$$

$$7.477\left(x_6^3 + x_7^3\right) + 0.7854\left(x_4 x_6^2 + x_5 x_7^2\right)$$

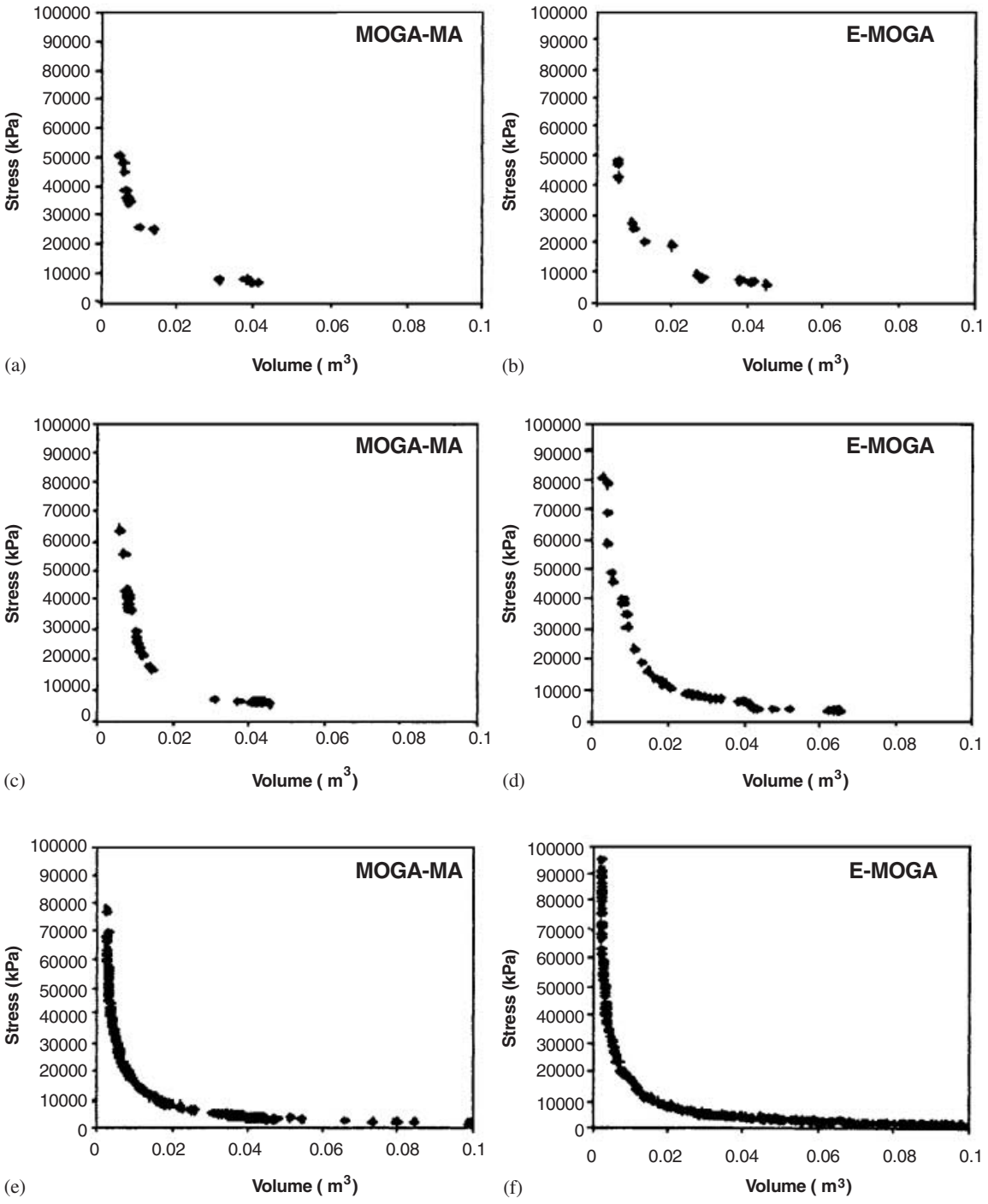


Fig. 8 Pareto solution sets for the two-bar truss example: (a) MOGA-NA; 150 function calls, (b) E-MOGA; 150 function calls, (c) MOGA-NA; 250 function calls, (d) E-MOGA; 250 function calls, (e) MOGA-NA; 550 function calls, (f) E-MOGA; 550 function calls

$$\text{Minimize} \quad f_{\text{stress}} = f_2 = \frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 1.69 \times 10^7}}{0.1x_6^3}$$

s.t.

$$g_1 : \frac{1}{x_1x_2^2x_3} - \frac{1}{27} \leq 0$$

$$g_2 : \frac{1}{x_1x_2^2x_3^2} - \frac{1}{397.5} \leq 0$$

$$g_3 : \frac{x_4^3}{x_2x_3x_6^4} - \frac{1}{1.93} \leq 0$$

$$g_4 : \frac{x_5^3}{x_2x_3x_7^4} - \frac{1}{1.93} \leq 0$$

$$g_5 : x_2x_3 - 40 \leq 0$$

$$g_6 : \frac{x_1}{x_2} - 12 \leq 0$$

$$g_7 : 5 - \frac{x_1}{x_2} \leq 0$$

$$g_8 : 1.9 - x_4 + 1.5x_6 \leq 0$$

$$g_9 : 1.9 - x_5 + 1.1x_7 \leq 0$$

$$g_{10} : f_1(\mathbf{x}) \leq 1300$$

$$g_{11} : \frac{\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 1.575 \times 10^8}}{0.1x_7^3} \leq 1100\,.$$

The lower and upper limits on the variables are

$$g_{12,13} : 2.6 \leq x_1 \leq 3.6$$

$$g_{14,15} : 0.7 \leq x_1 \leq 0.8$$

$$g_{16,17} : 17 \leq x_1 \leq 28$$

$$g_{18,19} : 7.3 \leq x_1 \leq 8.3$$

$$g_{20,21} : 7.3 \leq x_1 \leq 8.3$$

$$g_{22,23} : 2.9 \leq x_1 \leq 3.9$$

$$g_{24,25} : 5.0 \leq x_1 \leq 5.5$$

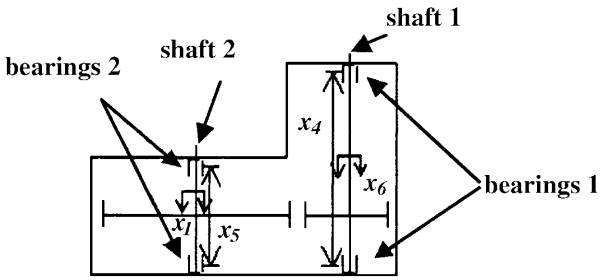


Fig. 9 Speed-reducer

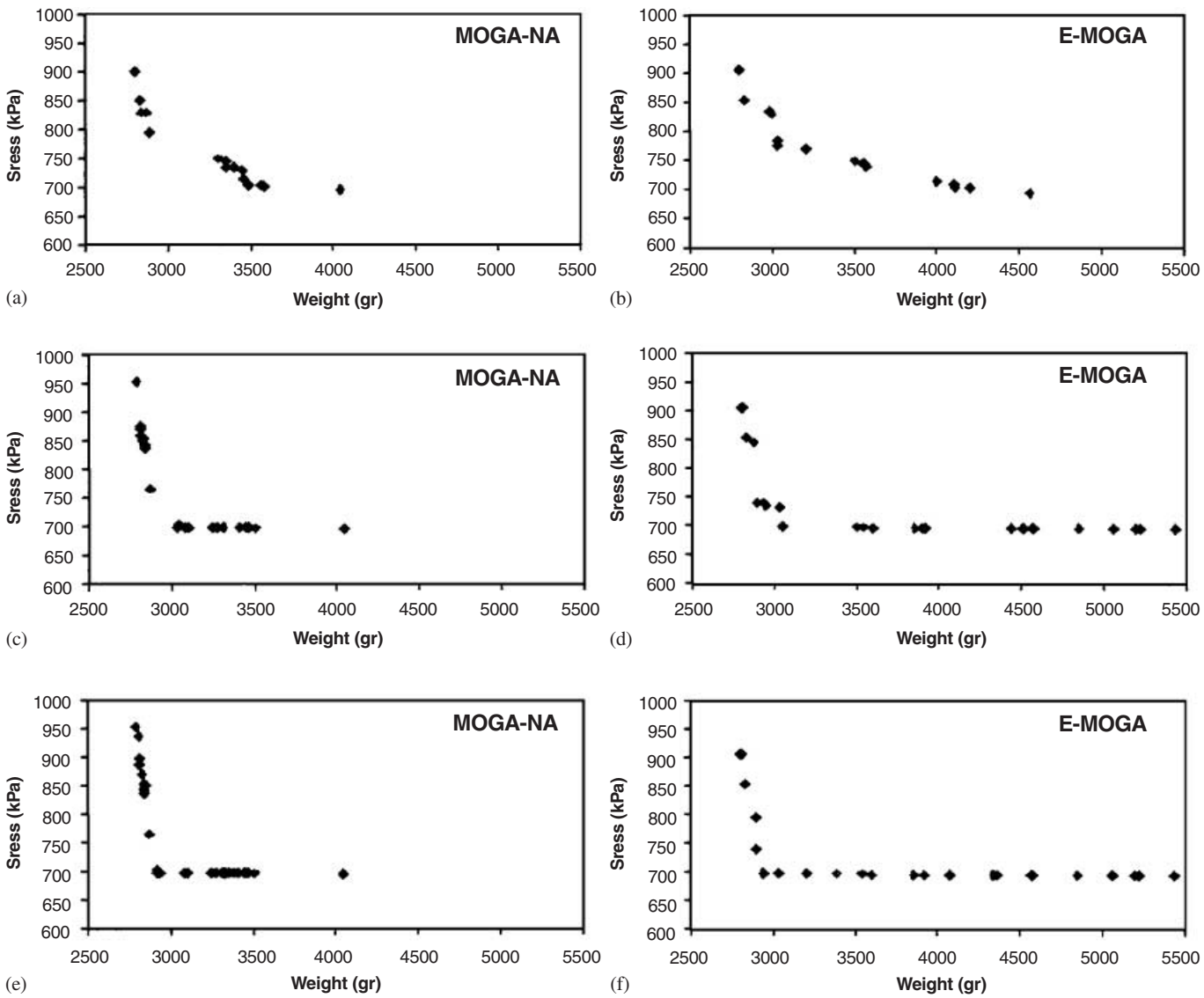


Fig. 10 Pareto solution sets for speed-reducer example: (a) MOGA-NA; 150 function calls, (b) E-MOGA; 150 function calls, (c) MOGA-NA; 250 function calls, (d) E-MOGA; 250 function calls, (e) MOGA-NA; 550 function calls, (f) E-MOGA; 550 function calls

Table 2 MOGA parameters in the speed-reducer example

Optimization technique	MOGA-NA	E-MOGA
population size	50	50
replacement per Generation	10	10
function calls	550	550
crossover type	2-point	2-point
crossover probability	0.8	0.8
mutation probability	0.05	0.05
bits per variable	10	10
expansion percentage	N/A	10%
expansion start	N/A	5-th gen.
expansion finish	N/A	45-th gen.

The Pareto solutions obtained using E-MOGA and MOG-NA are shown in Fig. 10 and the parameters are listed in Table 2. It is clear from these graphs that E-MOGA has again out-performed MOGA-NA, in terms of both the coverage of the Pareto frontier and the uniformity of spread. The range of pareto frontier covered by solutions from E-MOGA is significantly larger than that of MOGA-NA. Also, the solution points are spread uniformly in E-MOGA while they are mostly clustered in MOGA-NA.

4 Conclusions

In this paper, we presented a new multi-objective genetic algorithm, an entropy-based MOGA (E-MOGA), based on an analogy with the thermodynamic behavior of an ideal gas undergoing expansion in an enclosure. The E-MOGA expands a sample of population while it also the population evolves as it approaches the Pareto frontier in order to achieve maximum uniformity and coverage of the solutions. The constraint handling technique developed in E-MOGA is based on a feasible direction concept. That is, if individuals in the initial population satisfy the constraints, all of their subsequent generations will remain feasible.

As a demonstration, both E-MOGA and a baseline MOGA, i.e. MOGA-NA, were applied to a two-bar-truss example and a speed-reducer optimal design problem. In both examples, the results by E-MOGA showed dramatic improvement in terms of uniformity and coverage of solutions. In particular, the Pareto solutions obtained from MOGA-NA contained obvious gaps and clusters while those from E-MOGA were almost free of them. In addition, the solutions obtained from E-MOGA covered a significantly wider range of the Pareto frontier compared to MOGA-NA.

Acknowledgements The work presented here is supported in part by ONR contract N000149810842, and in part by NSF Grant 0112767. This support is gratefully acknowledged.

References

Azarm, S.; Tits, A.; Fan, M.K.H. 1989: Tradeoff driven optimization-based design of mechanical systems. *4-th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization*, pp. 551–558

Back, T. 1996: *Evolutionary algorithms in theory and practice.* New York, NY: Oxford University Press

Camponogara, E.; Talukdar, S. 1997: A genetic algorithm for constrained and multi-objective optimization. *3-rd Nordic Workshop on Genetic Algorithms and Their Applications*, pp. 49–62

Coello Coello, C.A. 1999: An updated survey of evolutionary multiobjective optimization techniques: State of the art and future trends. *Congr. on Evolutionary Computation*, pp. 3–13. Piscataway, NJ: IEEE Service Center

Cui, X.; Li., M.; Fang, T. 2001: Study of population diversity of multiobjective evolutionary algorithm based on immune and entropy principles. *Congress on Evolutionary Computation*, 2:1316–1321. Piscataway, NJ: IEEE Service Center

Desloge, E.A. 1966: *Statistical physics.* New York, NY: Holt Rinehard and Winston Inc.

Fay, J.A. 1965: *Molecular thermocynamics.* Reading, MA: Addison Wesley

Fonseca, C.M.; Fleming, P.J. 1995: An overview of evolutionary algorithms in multi-objective optimization. *Evolutionary Computation* **3**, 1–16

Kirsch, U. 1981: *Optimal structural design.* New York, NY: McGraw-Hill.

Kita, H.; Yabumoto, Y.; Mori, N.; Nishikawa, Y. 1996: Multi-objective optimization by means of the thermodynamical genetic algorithm optimization. *Lecture Notes in Computer Science*, pp. 504–512. Berlin, Heidelberg, New York: Springer

Narayanan, S.; Azarm, S. 1999: On improving multi-objective genetic algorithms for design optimization. *Struct. Optim.* **18**, 146–155

Penrose, O. 1970: *Foundations of statistical mechanics.* Oxford: Pergamon

Prugel-Bennett, A.; Shapiro, J. 1994: An analysis of genetic algorithms using statistical mechanics. *Phys. Rev. Lett.* **72**, 305–1309

Prugel-Bennett, A.; Shapiro, J.L. 1997: The dynamics of a genetic algorithm for simple random ising systems. *Physica D.* **104**, 75–114

Reynolds, B.J. 2000: *Hybrid multi-objective genetic algorithm with heuristics based on archived information.* M.S. Thesis, Dept. of Mechanical Engineering, University of Maryland

Shapiro, J.L.; Prugel-Bennett, A. 1995: Maximum entropy analysis of genetic algorithm operators. *Evolutionary Com-*

puting: AISB Workshop (held in Sheffield, UK), pp. 14–24. Berlin, Heidelberg, New York: Springer

Shapiro, J.L.; Prugel-Bennett, A.; Rattray, L.M. 1994: A statistical mechanical formulation of the dynamics of genetic algorithms. In: Fogarty, T.C. (ed.) *Lecture notes in computer science*, Vol. 865. Berlin, Heidelberg, New York: Springer

Sobieski, J.S.; Laba, K.; Kincaid, R. 1998: Bell-curved based evolutionary optimization algorithms. *7-th AIAA/USAF/NASA/ISSMO Symp. on Multidisciplinary Analysis and Optimization*, pp. 2083–2096

Toda, M.; Kubo, R.; Saito, N. 1991: *Statistical physics.* Berlin, Heidelberg, New York: Springer