COMPARATIVE STUDY OF SEVERAL MULTI-OBJECTIVE GENETIC ALGORITHMS

Ali Farhang Mehr, Ph.D.
Research Scientist
Intelligent Systems Division
NASA Ames Research Center
Moffett Field, CA 94035 USA
Phone: (650) 604-1140
E-mail: amehr@email.arc.nasa.gov

ABSTRACT

Among various optimization techniques in the literature, Genetic Algorithms (GA's) have received significant attention in the engineering design optimization domain. This is often attributed to GA's ability to handle problems with mixed discrete-continuous design variables as well as its straightforward generalization to multi-objective optimization. As a result, researchers have developed a variety of Multi-Objective GA's (MOGA's) over the last few years, each claiming to perform better than others in one aspect or another. The relative performance of these MOGA's in various engineering problem domains, however, remains for the most part understudied.

As such, in this paper, we investigate the relative performance of several well-known Multi-Objective GA's (MOGA's) on a quantitative and objective basis. Two aspects of MOGA performance are studied and compared: 1) convergence rate to the Pareto frontier; and 2) diversity of solutions. Two quantitative measures (i.e. quality indices) are selected accordingly, each addressing one aspect of performance. To account for the stochastic nature of GA's, multiple runs are performed for each MOGA to create samples of solution sets for statistical comparison. Based on the results of this study, we will observe several characteristics of the tested MOGA's, which can be helpful in selecting an appropriate MOGA for a given multi-objective optimization task.

Categories and Subject Descriptors
I.2 [**Artificial Intelligence**]

General Terms
Algorithms, Performance, Measurement.

Keywords
Multi-objective Genetic Algorithms; Design Optimization; Quality Indexes; Comparative Study.

1. INTRODUCTION

Since GA's are population-based search techniques, they can be easily modified to find an estimate of the entire Pareto frontier in a single run (For a review of MOGA's see for instance, Foneseca and Fleming 1993; Deb 2001; Coello Coello et al. 2002). In fact, many researchers (e.g., Valenzuela-Rendon and Uresti-Charre 1997) suggest that multi-objective optimization is a problem area where GA's perform better than other techniques. Although this statement is subjective and many conditions may have to hold (Wolpert and Macready 1997), nevertheless, Multi-Objective GA's (or MOGA's) are becoming quite popular among researchers in engineering and other domains.

A rapid growth in the complexity of modern day's computer-based engineering design problems has raised a need for more efficient MOGA's. Recently developed MOGA's are expected to converge faster and obtain a more diverse set of design solutions compared to earlier versions of these algorithms. Accordingly, researchers have developed a variety of different MOGA's; each claiming to perform better than others in one aspect or another (see Foneseca and Fleming 1995; Horn 1997; Zitzler 1999; Van Veldhuizen and Lamont 2000; Deb 2001; Coello Coello et al. 2002). While it is highly desirable to find a MOGA that performs better than others in all aspects and for all optimization problems, such a comprehensive technique has yet to be developed. In fact, Bosman and Thierens (2003) argue that a MOGA that performs well in a certain aspect such as fast convergence often tends to perform poorly in another aspect such as diversity of solutions. Therefore, a comparative study of different MOGA's is extremely important for engineering optimization practice, since design optimization problems with different characteristics may require different MOGA techniques.

In this paper, several recent as well as earlier versions of MOGA's are chosen for a comparative study. A comparison scheme is devised to compare these MOGA's in terms of: 1) fast convergence; and 2) maintaining diversity in the solution set. Since intuitive or visual comparison of solutions (i.e., Pareto frontier) from these MOGA's is often misleading, the proposed comparison scheme utilizes two quality indexes (as defined in Section 2), each measuring one of the above aspects of performance on a quantitative basis. It is argued that using these two quality indexes reduces complexity of the comparative study while maintaining its comprehensiveness (with respect to both aspects of quality). To account for random nature of GA's, a random sample of solution sets are obtained for each technique (from different random initial populations). The conclusions of this comparative study will be based on median and range of these statistical samples. Although in some instances the comparison of these MOGA's remains inconclusive, several observations are made that can be used in selection of appropriate MOGA's for a given design optimization task.

The organization of the rest of this paper is as follows: Section 1.1 presents the terminology of the paper. We provide an overview of MOGA's that are used in the study in Section 1.2. Section 2 explains the fundamentals of comparing MOGA's using quality indexes. We introduce two quality indexes that will be used later in Section 3 as the basis of our comparative study. In Sections 3.1 and 3.2, we present our engineering design optimization test suite: design optimization of a vibrating platform and a speed-reducer gearbox. In Section 3.3, the two quality indexes are used to compare the tested MOGA's. Several observations and recommendations are made accordingly. Finally, the concluding remarks of the paper are given in Section 4.

1.1 TERMINOLOGY

A multi-objective optimization problem with m objective functions ($m > 1$) can be shown in the following minimization form.

$$\begin{aligned} &\textit{Minimize} \quad \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}),\ldots,f_i(\mathbf{x}),\ldots,f_m(\mathbf{x})\} \\ &\text{subject to:} \quad \mathbf{x} \in D \qquad (1) \\ &D = \{\mathbf{x} : g_j(\mathbf{x}) \le 0, j = 1,\ldots,J; h_k(\mathbf{x}) = 0, k = 1,\ldots,K\} \end{aligned}$$

where $\mathbf{x}$ is an n-dimensional design variable vector, and D is the set of all such vectors that satisfy the constraints: g_j's and h_k's are the inequality and equality constraints, respectively. In the following, we define a few terms used in this paper.

Dominance: Let $\mathbf{x}_1$, $\mathbf{x}_2$ be two feasible design points. Then, $\mathbf{x}_1$ dominates $\mathbf{x}_2$ iff $f_i(\mathbf{x}_1) \le f_i(\mathbf{x}_2)$ for all i=1,…, m, with strict inequality for at least one i (Steuer, 1986).

Pareto-Optimal Solution: A feasible solution point, namely $\mathbf{x}^* \in D$, is Pareto optimal iff there does not exist another solution point, $\mathbf{x} \in D$, such that $f_i(\mathbf{x}) \le f_i(\mathbf{x}^*)$ for i=1,…,m, with strict inequality for at least one i (Steuer, 1986; Miettinen, 1999).

Pareto Frontier: The set of all Pareto optimal solutions to multi-objective optimization problem of Equation 1 is referred to as the Pareto frontier.

Normalization of Objectives: The objective functions of a multi-objective optimization problem are usually incommensurable in the sense that they have different units and therefore any comparison or aggregation among them is meaningless. To address this issue, the objectives are often normalized (Miettinen, 1999) with respect to two reference points: The *ideal* and *nadir* points, as defined in the following.

Ideal/good point: The ideal point is defined as a point in the objective space, whose components are obtained by constrained minimization of each of the objective functions individually, that is:

$$\begin{aligned} &\textit{Minimize} \ f_i(\mathbf{x}) \\ &\text{subject to: } \mathbf{x} \in D; \text{ for } i=1,\ldots,m \qquad (2) \end{aligned}$$

In practice, however, performing several optimization routines to obtain the ideal point is time-consuming. In most cases, an experienced engineer is able to estimate this ideal point even without optimizing the objectives. In this paper we refer to the ideal point or its best estimate as a *good point*. The good point is basically a lower bound for all objectives and should be selected such that it dominates all solution points.

Nadir/bad point: The nadir point is the opposite of the ideal point, i.e., the upper bounds of the Pareto frontier (Miettinen, 1999). Finding the nadir point is still an open research problem in general. There are a few attempts in the literature to further improve the estimation of the nadir point, but most of them require several optimization routines (see, for instance, Korhonen and Steuer, 1997). Instead, in this paper, we arbitrarily overestimate the ranges of objectives such that no design point is encountered that violates the estimated upper bounds. These estimated upper bounds for the objective functions constitute a point in the objective space that is referred to as the bad point. All solution points are normalized in the objective space with respect to these two good and bad reference points.

Non-Dominated Set (NDS): As shown in Figure 1, population-based multi-objective optimization techniques usually generate a finite set of design points to the optimization problem of Equation 1. If we denote the population of all feasible design points by S, then an NDS is defined as the set of all $\mathbf{x} \in S$ such that there does not exist another design point in S that dominates $\mathbf{x}$. Note that a design solution point in an NDS is not necessarily Pareto optimal. However, a good optimization algorithm will provide an NDS that approximates the Pareto frontier as closely as possible.

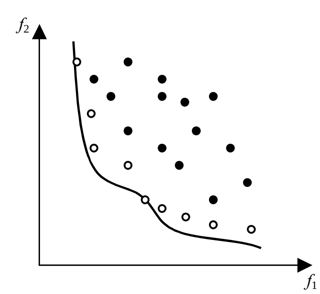


Figure 1: A typical population of solutions generated by a MOGA for a two-objective minimization problem. Hollow circles mark the Non-Dominated Set (NDS).

1.2 OVERVIEW OF MOGA'S

In general, MOGA's can be categorized into two main categories: weighting approaches, and population-based approaches (Wu and Azarm, 2001). In the first category, all objectives are combined into a single objective form using a set of weights and then a single-objective GA is applied to find a Pareto-optimal solution. Despite the simplicity of this approach, its applications are limited since the appropriate objective-weights are not known prior to the optimization process. Moreover, the weighted approach does not necessarily yield the entire Pareto-optimal solutions for non-convex problems. In the population-based approach, in contrast, a population of solutions is evolved simultaneously throughout the process to create a set of Pareto optimal solutions. Vector Evaluated GA (VEGA) developed by Schaffer (1985) is perhaps the most well-known technique in this category. Goldberg (1989) suggested another population-based approach by assigning a Pareto fitness value to the individual solutions from a multi-objective GA according to their dominance number. This approach soon became widely accepted and many MOGA's were developed based on this idea (e.g., Fonseca and Fleming 1993, Narayanan and Azarm 1999, among many others). Unlike classical techniques that require weighting, ε-constraint or other methods to transform multiple objectives into a single scalar, MOGAs are capable of handling multiple objectives, in most cases only by redefining the fitness criteria.

We selected several well-known or relatively recent MOGA's for the comparative study of this paper[1]:

- Vector Evaluated GA (VEGA; Schaffer 1985)
- Fonseca and Fleming's MOGA (denoted by FF-MOGA in this paper; Fonseca and Fleming 1993)
- Narayanan & Azarm's MOGA (MOGA-NA; Narayanan and Azarm 1999)
- Non-dominated Sorting Genetic Algorithm (NSGA; Deb 2001)
- Strength Pareto Evolutionary Algorithm (SPEA; Zitzler and Thiele, 1999)
- Entropy-based MOGA (E-MOGA, Farhang-Mehr and Azarm 2002)

Note that other than the above mentioned approaches, many other successful implementations of MOGA's are reported in the literature (e.g., Foneseca and Fleming 1995; Horn 1997; Zitzler 1999; Deb 2001; Van Veldhuizen and Lamont 2000; Coello Coello et al. 2002). The general goal of all of these MOGA's, however, is the same: To find a finite solution set that approximates the Pareto frontier as closely as possible. Nevetheless, obtaining the 'best possible' solution set is not always a trivial or objectively-defined task. Researchers have developed a myriad of MOGA's in the past few years that aim at improving the quality of the obtained solution sets in one way or another. However, it is often very difficult to determine how much these techniques have been successful in achieving this task. As such, quantitative measures of quality, referred to as *quality indexes*, have been developed that can be used to assess and compare the performance of MOGA's on a quantitative basis. The next section is devoted to this issue.

2. COMPARATIVE STUDY OF MOGA'S

Along with advances in the development of more sophisticated MOGA's, performance assessment and comparative study of such techniques also gained much attention (see, for instance, Zitzler and Thiele 1989; Van Veldhuizen 1999; Sayin 2000; Wu and Azarm 2001; Knowles and Corne 2002; and Zitzler et al. 2003). The most common way to compare different MOGA's is to simply visualize the NDS obtained from each technique and accordingly judge the superiority of one technique to another. As discussed by Van Valdhuizen and Lamont (2000), however, visual assessment is not a reliable tool for comparison of different multi-objective optimization techniques. Particularly, for problems with three or more design objectives, visual judgment is either impossible or quite misleading. A quality index, on the other hand, assigns an absolute or relative value to a non-dominated solution set to determine whether it is a 'good' representation of the Pareto frontier:

Quality Index: If A and B are two NDS's obtained from two different MOGA's, then a quality index, $Q(A,B)$, provides a scalar that reflects how much better set A is than set B with respect to a certain aspect of quality.

Since quality of an NDS set is not a well-defined or well-understood concept, the common trend among researchers is to *decompose* the general notion of quality into several aspects and develop quality indexes accordingly. For instance, Deb (1998) states that MOGA's must perform well in two aspects:

1) *convergence to the Pareto frontier*
2) *maintaining diversity of solutions*

Bosman and Thierens (2003) also suggested similar decomposition and stated that there is a tradeoff between these two aspects in most cases, although MOGA's that perform better with respect to both are feasible. Several quality indexes have been developed to address each of the above aspects of performance. Examples of diversity quality indexes include: 'spacing index' (Schott, 1995); 'overall non-dominated vector generation', 'overall non-dominated vector generation ratio' (Van Veldhuizen 1999); 'coverage', 'uniformity', 'cardinality' (Sayin 2000); 'number of distinct choices', 'Pareto spread', and 'cluster' (Wu and Azarm 2001). In a similar fashion, researchers developed numerous indexes to assess the closeness of solution sets to the Pareto frontier (see Knowles 2002, for examples of these indexes).

Obviously, many of indexes that address a common aspect of quality are correlated in one way or another, introducing redundancy in the comparison study of MOGAs. Therefore, selecting too many of these indexes is not only confusing but also transforms the quality assessment of MOGAs into unnecessary and complex tradeoffs among different indexes. In fact, it is often impossible to find a situation in which a MOGA outperforms other MOGAs in terms of all existing quality indexes. That is, one algorithm for example may produce more distinct solutions while the other distributes the solutions more uniformly and a third one performs better in terms of having no gaps among the solution points. Having the correlation among different quality indexes in mind, Farhang-Mehr and Azarm (2003a) proposed using only two quality indexes: one representing the convergence to the Pareto frontier, and the other representing the diversity of solution points. For the former, they suggested using *Size of the Dominated Space* (S index, Zitzler 1999), and for the latter they formulated a new quality index, referred to as the *Entropy Index* (H index, Farhang-Mehr and Azarm 2003b). They showed that these two indexes (described briefly in the following) have

[1] These MOGA's are implemented by the author of this paper, based on the descriptions given in the corresponding references.

negligible statistical correlation and add minimum redundancy in the comparative study.

Size of Dominated Space (S index): This quality index provides a measure of convergence (or optimality) of a given NDS: the closer the solution set gets to the Pareto frontier, the lower the value of S index becomes (Fonseca and Fleming 1995; Zitzler 1999). Consider a non-dominated minimization solution set: $A=\{\mathbf{x}_1,\ldots,\mathbf{x}_i,\ldots\}$ in a normalized design objective space (with the ideal point transformed to the origin). The size of the dominated space by set A, denoted by $S(A)$, is defined as the volume of the union of hypercubes $\{C_1,\ldots,C_i,\ldots\}$, where C_i is a hypercube whose two opposite vertices are $\mathbf{x}_i$ and the origin of the objective space. Figure 2, for instance, shows the two hypercubes generated by the non-dominated set $A=\{\mathbf{x}_1, \mathbf{x}_2\}$. The volume of the union of these two hypercubes measures $S(A)$.

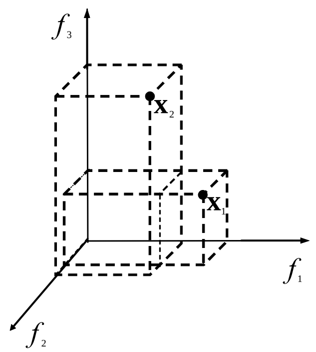


Figure 2: Size of dominated space, S index, for the set $\{\mathbf{x}_1, \mathbf{x}_2\}$. (A lower S value indicates a better convergence.)

Entropy (H index): Just as it is important to converge to the Pareto frontier, it is equally important to maintain diversity in the solution set. In MOGA's, in particular, the solutions tend to converge into clusters leaving the rest of the Pareto frontier empty or sparsely populated, a phenomenon known as *genetic drift* (Goldberg, 1989). Farhang-Mehr and Azarm (2003b) developed an entropy-based quality index that can be used to assess the 'diversity' of a NDS. The basic idea behind the entropy index is that each solution point provides some information about its neighborhood that can be modeled as a function, called an influence function. Figure 3, for instance, shows several solution points along a line segment. The impact of each solution point on its neighborhood is modeled by a Gaussian influence function. The influence function of the i-th solution point is maximum at that point and decreases gradually with the distance. Now, the density function, d, is defined as the aggregation of the influence functions from all solution points.

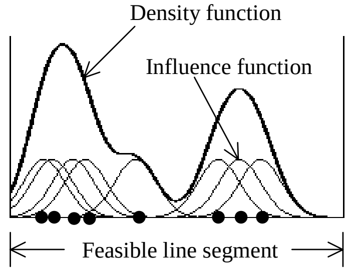


Figure 3: A set of solution points in a one-dimensional feasible space with the corresponding influence and density functions.

The density curve of Figure 3 (or density hyper-surface for multiple dimensions) consists of peaks and valleys. The peaks correspond to dense areas with many solution points in the vicinity and the valleys correspond to sparse areas with few adjacent points. A desirable solution set must have a 'flat' density surface. To quantify this flatness, one may take advantage of the similarities between this problem and the Shannon's entropy in information theory, which also measures the flatness of a distribution. This can by done by constructing a mesh in the solution hyper-plane and normalizing the values of density function, measured at the nodes:

$$\rho_i = \frac{d_i}{\sum_i d_i} \tag{3}$$

where d_i represents the density function at the i-th node. Now we have:

$$\sum_i \rho_i = 1 \qquad ; \rho_i \geq 0 \tag{4}$$

The entropy of such a distribution can then be defined as:

$$H = -\sum_i \rho_i ln(\rho_i) \tag{5}$$

A set of uniformly distributed solutions yields a relatively even surface without significant peaks and valleys. This corresponds to a high entropy index H. In contrast, if the solution points are grouped into one or more clusters, leaving the rest of the area sparsely populated, the density function contains sharp peaks and deep valleys, which corresponds to a lower entropy index. This provides a quantitative measure for the spread of a NDS over the Pareto frontier: higher H value indicates better diversity (See Farhang-Mehr 2003b).

In the following section, two engineering design optimization problems are introduced: 1) a vibrating platform; and 2) a speed reducer gearbox. The performance of the selected MOGA's (recall Section 2) are tested for these problems using S and H indexes. As discussed before, these quality indexes measure the performance with respect to two aspects of quality: convergence and diversity, respectively.

3. COMPARATIVE STUDY USING ENGINEERING TEST PROBLEMS

Sections 3.1 and 3.2 introduce vibrating platform and speed reducer gearbox design problems. The H and S quality indexes are used in Section 3.3 to compare the performance of the tested MOGA's.

3.1 DESIGN OF A VIBRATING PLATFORM

This design problem was originally taken from Messac (1996) but modified to form a multi-objective optimization problem. The design consists of a pinned-pinned sandwich beam with a vibrating motor on its top. As shown in Figure 4, the beam has five layers of three different materials. There is a middle layer and two sandwiched layers. The distance from the center of the beam to the outer edge of each layer comprises three of the sizing design variables, d_1, d_2, and d_3. The width of the beam, b, and the length of the beam, L, are the other two sizing design variables. There are also three combinatorial variables for the material type M_i, where i=1,2,3, for the different materials that can be used for each layer. Hence, there are 8 design variables, 3 combinatorial variables for the material type of the 3 layers, and 5 sizing variables.

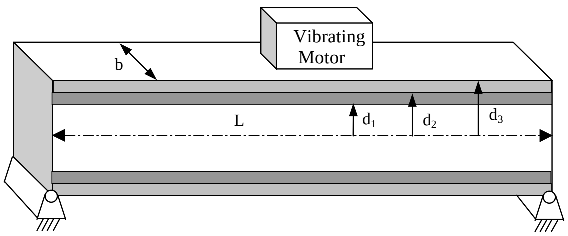


Figure 4: Vibrating platform design problem.

The problem has two design objectives: 1) Maximize the fundamental frequency of the beam, and 2) Minimize the material cost. The maximization of the first objective is converted to a minimization form by adding a negative sign to it. Therefore, the problem can be formulated as follows:

$$\text{Minimize} \quad f_1(d_1,d_2,d_3,b,L,M_i) = -(\pi/2L^2)(EI/\mu)^{0.5}$$
$$\text{Minimize} \quad f_2(d_1,d_2,d_3,b,M_i) = 2b[c_1 d_1 + c_2(d_2 - d_1) + c_3(d_3 - d_2)]$$

subject to:
$$\mu L - 2800 \le 0 \tag{6}$$
$$d_2 - d_1 - 0.15 \le 0$$
$$d_3 - d_2 - 0.01 \le 0$$
$$0.05 \le d_1 \le 0.5$$
$$0.2 \le d_2 \le 0.5$$
$$0.2 \le d_3 \le 0.6$$
$$0.35 \le b \le 0.5$$
$$3 \le L \le 6$$

where, (7)

$$(EI) = (2b/3)[E_1 d_1^3 + E_2(d_2^3 - d_1^3) + E_3(d_3^3 - d_2^3)]$$
$$(\mu) = 2b[\rho_1 d_1 + \rho_2(d_2 - d_1) + \rho_3(d_3 - d_2)]$$

Here, E_i is the modulus of elasticity of material M_i, while ρ_i is the density, and c_i is the cost. According to the material type variable M_i, the value of the parameters E_i, ρ_i, and c_i is different for different layer material, as given in Table 1. It is assumed that the material types for the three layers are mutually exclusive. In other words, the same material cannot be used for more than one layer. However, the layers are allowed to have zero thickness. The first three constraints refer to upper bounds on the mass of the beam, thickness of layer 2, and thickness of layer 3, respectively. The last 5 constraints are the set constraints on the sizing variables.

Table 1: Material properties of the vibrating platform

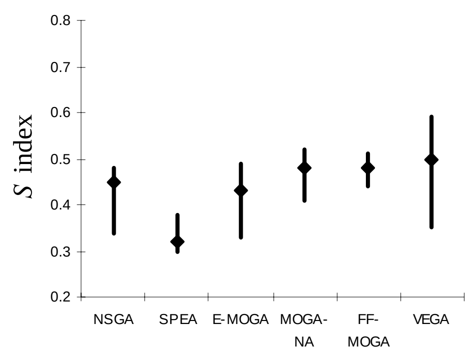
Material M_i	ρ_i (Kg/m^3)	E_i (N/m^2)	C_i (\$/volume)
1	100	1.6×10^9	500
2	2,770	70×10^9	1,500
3	7,780	200×10^9	800

Table 2 below lists the GA parameters used in this paper.

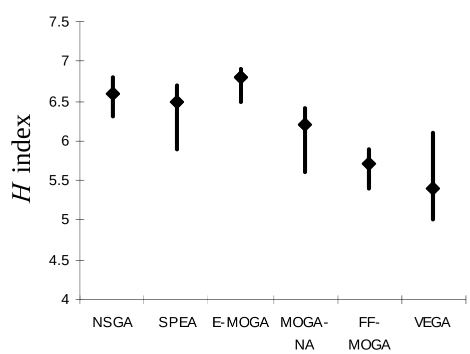
Table 2: GA parameters.

Parameter	Value
Population size	100
Replacement per generation	10
Function calls	550
Crossover type	2-point
Crossover probability	0.8
Mutation probability	0.05
Bits per variable	10
Selection type	Stochastic universal selection

Since GA is a stochastic process by nature, each MOGA is executed 50 times (with different randomly-generated initial populations and random seeds) and the final NDS's are stored for comparison. The best, the worst, and the median of each quality index (in the sample of 50 NDS's for each technique) are shown in Charts 1(a) and 1(b). Some observations and recommendations are made in Section 3.3 based on the results shown in Charts 1(a) and 1(b).



(a)



(b)

Chart 1: Median, minimum, and maximum values of the (a) *S* index, and (b) *H* index, for the tested multi-objective GA's (design of a vibrating platform).

3.2 SPEED REDUCER GEARBOX

This example is a modified version of a problem originally formulated by Golinski (1970) as a single objective optimization problem. Here, it has been converted to a three-objective optimization. The example represents the design of a simple gearbox, as shown in Figure 5. 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 1 (x_4), distance between bearings 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 other two objectives, f_2 and f_3, are to minimize the stress in the first and second shafts, respectively.

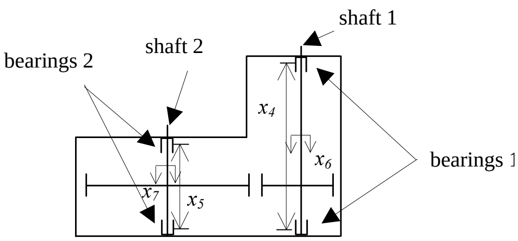


Figure 5: Speed gear reducer

The optimization formulation of the gear reducer has a number of constraints due to the gear and shaft design practices. These 11 inequality constraints are 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. Additionally, upper and lower limits are imposed on each of the seven design variables. The optimization formulation is:

$$\begin{aligned} \text{Minimize } f_{weight} = f_1 = {} & 0.7854 x_1 x_2^2 (10 x_3^2/3 + 14.933 x_3 \\ & - 43.0934) - 1.508 x_1 (x_6^2 + x_7^2) + \\ & 7.477 (x_6^3 + x_7^3) + 0.7854 (x_4 x_6^2 + x_5 x_7^2) \end{aligned}$$

$$\text{Minimize } f_{stress,1} = f_2 = \frac{\sqrt{(745 x_4 / x_2 x_3)^2 + 1.69x10^7}}{0.1 x_6^3}$$

$$\text{Minimize } f_{stress,2} = f_3 = \frac{\sqrt{(745 x_5 / x_2 x_3)^2 + 1.575x10^8}}{0.1 x_7^3}$$

Subject to: (8)

$$g_1 : \frac{1}{(x_1 x_2^2 x_3)} - \frac{1}{27} \le 0 \qquad g_2 : \frac{1}{(x_1 x_2^2 x_3^2)} - \frac{1}{397.5} \le 0$$

$$g_3 : \frac{x_4^3}{(x_2 x_3 x_6^4)} - \frac{1}{1.93} \le 0 \qquad g_4 : \frac{x_5^3}{(x_2 x_3 x_7^4)} - \frac{1}{1.93} \le 0$$

$$g_5 : x_2 x_3 - 40 \le 0 \qquad g_6 : \frac{x_1}{x_2} - 12 \le 0$$

$$g_7 : 5 - \frac{x_1}{x_2} \le 0 \qquad g_8 : 1.9 - x_4 + 1.5 x_6 \le 0$$

$$g_9 : 1.9 - x_5 + 1.1 x_7 \le 0 \qquad g_{10} : f_2(x) \le 1300$$

$$g_{11} : f_3(x) \le 1100$$

The lower and upper limits on the seven variables are:

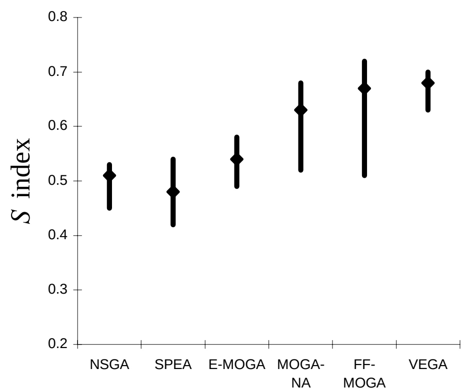
$$g_{12,13} : 2.6 \le x_1 \le 3.6 \qquad g_{14,15} : 0.7 \le x_2 \le 0.8$$

$$g_{16,17} : 17 \le x_3 \le 28 \qquad g_{18,19} : 7.3 \le x_4 \le 8.3$$

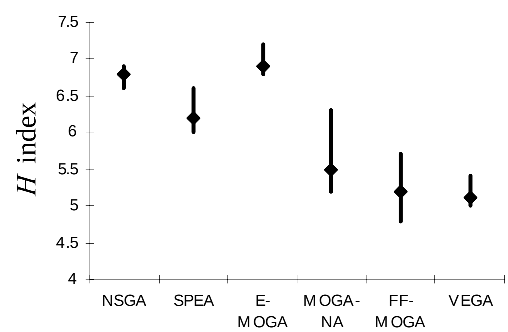
$$g_{20,21} : 7.3 \le x_5 \le 8.3 \qquad g_{22,23} : 2.9 \le x_6 \le 3.9$$

$$g_{24,25} : 5.0 \le x_7 \le 5.5$$

Charts 2(a) and 2(b) demonstrate the best, the worst, and the median values for each tested MOGA in this example. Some observations and recommendations are made based on these results in the next section.



(a)



(b)

Charts 2: Median, minimum, and maximum values of the (a) *S* index, and (b) *H* index, for the tested multi-objective GA's (design of a speed-reducer gearbox).

3.3 OBSERVATIONS

Charts 1(a) and 1(b) summarize the range of *S* and *H* indexes obtained for the vibrating platform design problem.

Similarly, Charts 2(a) and 2(b) show the ranges for the speed-reducer gearbox problem. These charts can be used to make the following observations about the performance of the tested MOGA's in terms of convergence and diversity:

In terms of convergence: The medians of the *S* index for the tested MOGA's in Charts 1(a) and 2(b) show that the more recent MOGA's, i.e., NSGA, SPEA, and E-MOGA outperform other approaches by yielding lower *S* values in most cases. (Note that a lower *S* value indicates a better convergence.) Furthermore:

- SPEA in Chart 1(a) conclusively outperforms MOGA-NA and FF-MOGA (i.e., the entire range of *S* index from SPEA lies below the minimum value of *S* index from the other two approaches.).
- For gearbox design problem (Chart 2(a)), SPEA conclusively outperforms VEGA (i.e., the ranges of *S* index do not overlap).
- Other than the above conclusive comparisons, SPEA yields a lower median than any other MOGA in both cases. Nevertheless, the conclusions are not conclusive (since the ranges overlap).

In addition, since the median of *S* index from SPEA is less than the minimum *S* values from MOGA-NA, FF-MOGA, and VEGA in both Charts 1(a) and 2(a), it can be concluded that:

- SPEA converged better than MOGA-NA, FF-MOGA, and VEGA in more than 50% of cases.

Also,

- Although the median of SPEA is slightly lower than that of E-MOGA or NSGA, the comparison is inconclusive.

In terms of maintaining diversity: Again, the three more recent MOGA's, i.e., NSGA, SPEA, and E-MOGA obtained more diverse solution sets in most cases by yielding higher median *H* values. (Note that a higher *H* value indicates better diversity.) Furthermore,

- E-MOGA conclusively outperforms MOGA-NA, FF-MOGA, and VEGA in both problems (in terms of diversity).

E-MOGA conclusively outperforms SPEA in Chart 2(b). For the first test problem (Chart 1(b)), E-MOGA outperforms SPEA in more than 50% of the cases (since the median of the *H* index from E-MOGA is higher than maximum *H* value from SPEA. The comparison between E-MOGA and NSGA is inconclusive in both cases (although E-MOGA yields a higher median *H* value)

In addition, from comparing the median of *H* index from NSGA and SPEA with maximum *H* values from MOGA-NA, FF-MOGA, and VEGA, it can be concluded that:

- For both test problems (Charts 1(b) and 2(b)), NSGA resulted in a better diversity than MOGA-NA, FF-MOGA, and VEGA in more than 50% of the cases.
- Similarly, SPEA performed better than FF-MOGA, and VEGA in more than 50% of the cases. The comparison between SPEA and MOGA-NA remains inconclusive in Chart 2(b).

Recommendations: For both test problems, the more recent MOGA's, i.e., NSGA, SPEA, and E-MOGA, resulted in better median index values than older versions, i.e., MOGA-NA, FF-MOGA, and VEGA, in terms of both convergence and diversity. Among the more recent MOGA's, SPEA resulted in the best median *S* index for both test problems (faster convergence), while E-MOGA performed better in terms of achieving a higher median *H* value (better diversity). Based on these observations, it is recommended to employ SPEA where a faster rate of convergence is needed; while E-MOGA may be used for problems with features that hinder maximizing diversity in the solution set. NSGA, on the other hand, shows a balance between the above two tasks.

Finally, note that the above comparative study is based on the assumption that decision maker is interested in two aspects of quality, convergence and diversity. Obviously, different quality indexes must be chosen to address different issues of interest. The result of the above comparative study, therefore, depends on the decision-maker's notion of the concept of quality and may vary based on the selected quality indexes and test examples.

4. CONCLUDING REMARKS

In this paper, a comparative study was conducted to assess the relative performance of several MOGA's. The proposed study evaluated the tested MOGA's in two aspects: 1) convergence; and 2) maintaining diversity in the solution set. Two quality indexes were used to address each of the above two issues: *S* index, and *H* index, respectively. To account for the stochastic nature of MOGA's, we performed 50 runs for each tested MOGA (per each test problem) and cataloged the obtained NDS's. *S* and *H* indexes were evaluated for these NDS's. For each tested MOGA, minimum, maximum, and median index values were obtained. These ranges are then compared to observe the relative performance of the tested MOGA's. This comparison scheme has the following advantages:

- It is quantitative in the sense that it utilizes indexes instead of visual assessment.
- A separate quality index is used for each desired aspect of quality. This ensures a comprehensive assessment of the relative performances according to the index.
- It accounts for the stochastic nature of MOGA's by performing multiple runs (from different initial random populations).

In some instances, the statistical comparison of the tested MOGA's remained inconclusive. Nevertheless, several helpful observations were made that could be used to recommend an appropriate MOGA for a given design optimization task.

REFERENCES

Bosman, P. A. N., Thierens, D., 2003, "The Balance Between Proximity and Diversity in Multiobjective Evolutionary Algorithms," IEEE Transactions on Evolutionary Computation, Vol. 7(2), pp. 174-188.

Coello Coello, C. A., Van Veldhuizen, D. A., and Lamont, G. B., 2002, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, New York, NY.

Deb, K., 1998, *Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems*, Technical Report CI-49/98, Department of Computer Science/LS11, University of Dortmund, Germany.

Deb, K., 2001, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley and Sons, Chichester, UK.

Farhang-Mehr, A., Azarm, S., 2002, "Entropy-Based Multi-Objective Genetic Algorithm for Design Optimization", Multidisciplinary and Structural Optimization, Vol 24(5), pp. 351-361.

Farhang-Mehr, A., and Azarm, S., 2003a, "Minimal Sets of Quality Metrics," Lecture Notes in Computer Science, Springer-Verlag Heidelberg, Vol. 2632, pp. 405-417.

Farhang-Mehr, A., and Azarm, S., 2003b, "An Information-Theoretic Performance Metric for Quality Assessment of Multi-Objective Optimization Solution Sets", Trans. of the ASME, Journal of Mechanical Design, Vol. 125(4), pp. 655-663.

Fonseca, C. M., Fleming, P. J., 1993, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization", The Fifth International Conference on Genetic Algorithms, San Mateo, CA, pp. 416-423.

Fonseca, C. M., Fleming, P. J., 1995, "An Overview of Evolutionary Algorithms in Multiobjective Optimization", Evolutionary Computation, Vol. 3(1), pp. 1-16.

Goldberg, D. E., 1989, *Genetic Algorithms in Search, Optimization and Machine Learning,* Addison-Wesley, Reading, MA.

Golinski, J., 1970, "Optimal Synthesis Problems Solved by Means of Nonlinear Programming and Random Methods", Journal of Mechanisms, Vol. 5, pp. 287-309.

Holland, J.H., 1975, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, MI.

Horn, J., 1997, *Multicriteria Decision Making*, in *Handbook of Evolutionary Computation*, Institute of Physics Publishing, Bristol, UK.

Knowles, J. D., Corne, D. W., 2002, "On Metrics for Comparing Non-Dominated Sets", Proceedings of the 2002 Congress on Evolutionary Computation Conference, Honolulu, Hawaii, pp. 711-716.

Knowles, J. D., 2002, *Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization*, Ph.D. Dissertation, University of Reading, Department of Computer Science, Reading, UK.

Korhonen, P. S., Steuer, R. E., 1997, "A Heuristic for Estimating Nadir Criterion Values in Multiple Objective Linear Programming," Operations Research, Vol. 45(5), pp. 751-757.

Messac, A., 1996, "Physical Programming: Effective Optimization for Computational Design," AIAA Journal, Vol. 34 (1), pp. 1490158.

Miettinen, K. M., 1999, *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers, Boston, MA.

Narayanan, S., Azarm, S., 1999, "On Improving Multiobjective Genetic Algorithms for Design Optimization", Structural Optimization, Vol. 18, pp. 146-155.

Sayin, S., 2000, "Measuring the Quality of Discrete Representations of Efficient Sets in Multiple Objective Mathematical Programming", Mathematical Programming, Vol. 87, pp. 543-560.

Schaffer, J. D., 1985, "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms", Genetic Algorithms and Their Applications, Proceedings of the First International Conference on Genetic Algorithms, Hillsdale, NJ, pp. 93-100.

Schott, J. R., 1995, *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization*, M.S. Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

Steuer, R. E., 1986, *Multiple Criteria Optimization: Theory, Computation, and Application*, John Wiley & Sons, New York

Valenzuela-Rendon, M., Uresti-Charre, E., 1997, "A Non-Generational Genetic Algorithm for Multiobjective Optimization", Proceedings of the Seventh International Conference on Genetic Algoriths, San Francisco, CA, pp. 658-665.

Van Veldhuizen, D. A., 1999, *Multiobjective Evolutionary Algorithm: Classification, Analyses and New Innovations*, Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

Van Valdhuizen, D.A., Lamont G.B., 2000, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art", Evolutionary Computation, Vol. 8(2), pp. 125-147.

Wolpert, D. H., Macready, W. G., 1997, "No Free Lunch Theorems for Optimization", IEEE Transactions on Evolutionary Computation, Vol. 1(1), pp. 67-82.

Wu, J., and S. Azarm, 2001, "Metrics for Quality Assessment of a Multiobjective Design Optimization Solution Set", Transactions of the ASME, Journal of Mechanical Design, Vol. 123, pp. 18-25.

Zitzler, E., Thiele, L., 1998, "Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Study", Proc. 5th International Conference, Parallel Problem Solving from Nature, Amsterdam, The Netherlands, pp. 292-301.

Zitzler, E., 1999, *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, Ph.D. Thesis, Swiss Federal Institute of Technology (ETH) Zurich.

Zitzler, E., Thiele, L., 1999, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach", IEEE Transactions on Evolutionary Computation, Vol. 3(4), pp. 257–271.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., Fonseca, V., 2003, "Performance Assessment of Multiobjective Optimizers: An Analysis and Review," IEEE Transactions on Evolutionary Computation, Vol. 7(2), pp. 117-132.