An Updated Survey of Evolutionary Multiobjective Optimization Techniques : State of the Art and Future Trends

Carlos A. Coello Coello
Laboratorio Nacional de Informática Avanzada
Rébsamen 80, Xalapa, Veracruz 91090, México
ccoello@xalapa.lania.mx

Abstract- This paper reviews some of the most popular evolutionary multiobjective optimization techniques currently reported in the literature, indicating some of their main applications, their advantages, disadvantages, and degree of aplicability. Finally, some of the most promising areas of future research are briefly discussed.

1 Introduction

Since the pioneering work of Rosenberg in the late 1960s regarding the possibility of using genetic-based search to deal with multiple objectives [1], this new area of research (now called Evolutionary Multi-Objective Optimization, or EMOO for short) has grown considerably as indicates the notable increment (mainly in the last 15 years) of technical papers in international conferences and peer-reviewed journals, special sessions in international conferences and interest groups in the Internet[1].

Multiobjective optimization is with no doubt a very important research topic both for scientists and engineers, not only because of the multiobjective nature of most real-world problems, but also because there are still many open questions in this area. In fact, there is not even a universally accepted definition of "optimum" as in single-objective optimization, which makes it difficult to even compare results of one method to another, because normally the decision about what the "best" answer is corresponds to the so-called (human) *decision maker*.

Evolutionary algorithms seem particularly suitable to solve multiobjective optimization problems because they deal simultaneously with a set of possible solutions (the so-called population) which allows to find an entire set of Pareto optimal solutions in a single run of the algorithm, instead of having to perform a series of separate runs as in the case of the traditional mathematical programming techniques. Additionally, evolutionary algorithms are less succeptible to the shape or continuity of the Pareto front, whereas these two issues are a real concern for mathematical programming techniques.

In this paper we will try to provide a quick review of the most important work performed in this area, indicating some of the current research trends and the most important areas of future research.

[1]The author maintains a list on Evolutionary Multiobjective Optimization at: `http://www.lania.mx/˜ccoello/EMOO/EMOObib.html`

2 Statement of the Problem

Multiobjective optimization (also called multicriteria optimization, multiperformance or vector optimization) can be defined as the problem of finding [2]:

> a vector of decision variables which satisfies constraints and optimizes a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria which are usually in conflict with each other. Hence, the term "optimize" means finding such a solution which would give the values of all the objective functions acceptable to the designer.

Formally, we can state it as follows:

We want to find the vector $\bar{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]^T$ which will satisfy the m inequality constraints:

$$g_i(\bar{x}) \geq 0 \quad i = 1, 2, \ldots, m \tag{1}$$

the p equality constraints

$$h_i(\bar{x}) = 0 \quad i = 1, 2, \ldots, p \tag{2}$$

and optimizes the vector function

$$\bar{f}(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), \ldots, f_k(\bar{x})]^T \tag{3}$$

where $\bar{x} = [x_1, x_2, \ldots, x_n]^T$ is the vector of decision variables.

In other words, we wish to determine from among the set $\mathcal{F}$ of all numbers which satisfy (1) and (2) the particular set $x_1^*, x_2^*, \ldots, x_k^*$ which yields the optimum values of all the objective functions.

3 Pareto Optimum

The concept of *Pareto optimum* was formulated by Vilfredo Pareto in the XIX century [3], and constitutes by itself the origin of research in multiobjective optimization. We say that a point $\bar{x}^* \in \mathcal{F}$ is *Pareto optimal* if for every $\bar{x} \in \mathcal{F}$ either,

$$\bigwedge_{i \in I} (f_i(\bar{x}) = f_i(\bar{x}^*)) \tag{4}$$

or, there is at least one $i \in I$ such that

$$f_i(\bar{x}) > f_i(\bar{x}^*) \tag{5}$$

In words, this definition says that $\bar{x}^*$ is Pareto optimal if there exists no feasible vector $\bar{x}$ which would decrease some criterion without causing a simultaneous increase in at least one other criterion. Unfortunately, the Pareto optimum almost always gives not a single solution, but rather a set of solutions called *non-inferior* or *non-dominated* solutions.

3.1 Pareto Front

The minima in the Pareto sense are going to be in the boundary of the design region, or in the locus of the tangent points of the objective functions. In Figure 1, a bold line is used to mark this boundary for a biobjective problem. The region of points defined by this bold line is called the *Pareto Front*. In general, it is not easy to find an analytical expression of the line or surface that contains these points, and the normal procedure is to compute the points $\mathcal{F}^k$ and their corresponding $f(\mathcal{F}^k)$. When we have a sufficient amount of these, we may proceed to take the final decision.

4 Some of the Most Important Approaches

Due to obvious space limitations, we cannot enumerate here all the different EMOO approaches that have been proposed in the literature[2], and we will limit our study to the approaches that have been more popular among researchers:

- Aggregating functions
- Schaffer's VEGA
- Fonseca and Fleming's MOGA
- Srinivas and Deb's NSGA
- Horn and Nafpliotis' NPGA
- Target vector approaches

4.1 Aggregating functions

Knowing that a genetic algorithm needs scalar fitness information to work, it is almost natural to propose a combination of all the objectives into a single one using either an addition, multiplication or any other combination of arithmetical operations that we could devise. An example of this approach is a sum of weights of the form:

$$\min \sum_{i=1}^{k} w_i f_i(\bar{x}) \tag{6}$$

where $w_i \geq 0$ are the weighting coefficients representing the relative importance of the objectives. It is usually assumed that

$$\sum_{i=1}^{k} w_i = 1 \tag{7}$$

[2]The interested reader should refer to [4, 5] for more detailed surveys of EMOO approaches.

4.1.1 Applications

Syswerda and Palmucci [6] used weights in their fitness function to add or subtract values during the schedule evaluation of a resource scheduler, depending on the existence or absence of penalties (constraints violated). Jakob et al. [7] used a weighted sum of the several objectives involved in a task planning problem : to move the tool center point of an industrial robot to a given location as precisely and quickly as possible, avoiding certain obstacles and aiming to produce a path as smooth and short as possible. Jones et al. [8] used weights for their genetic operators in order to reflect their effectiveness when a GA was applied to generate hyperstructures from a set of chemical structures. Wilson & Macleod [9] used this approach as one of the methods incorporated into a GA to design multiplierless IIR filters in which the two conflicting objectives were to minimize the response error and the implementation cost of the filter. Liu et al. [10] used this technique to optimize the layout and actuator placement of a 45-bar plane truss in which the objectives were to minimize the linear regulator quadratic control cost, the robustness and the modal controllability of the controlled system subject to total weight, asymptotical stability and eigenvalues constraints. Yang and Gen [11] used a weighted sum approach to solve a bicriteria linear transportation problem. More recently, Gen et al. [12, 13] extended this approach to allow more than two objectives, and added fuzzy logic to handle the uncertainty involved in the decision making process. A weighted sum is still used in this approach, but it is combined with a fuzzy ranking technique that helps to identify Pareto solutions, since the coefficients of the objectives are represented with fuzzy numbers reflecting the existing uncertainty regarding their relative importance.

4.1.2 Strengths and weaknesses

This method was the first technique developed for the generation of non-inferior solutions for multiobjective optimization. This is an obvious consequence of the fact that it was implied by Kuhn and Tucker in their seminal work on numerical optimization [14]. The main strength of this method is its efficiency (computationally speaking), and its suitability to generate a strongly non-dominated solution that can be used as an initial solution for other techniques. Its main weakness is the difficulty to determine the appropriate weights that can appropriately scale the objectives when we do not have enough information about the problem, particularly if we consider that any optimal point obtained will be a function of such weights. Still more important is the fact that this approach does not generate proper Pareto optimal solutions in the presence of non-convex search spaces regardless of the weights used [15].

4.2 Schaffer's VEGA

David Schaffer [16] extended Grefenstette's GENESIS program [17] to include multiple objective functions. Schaffer's

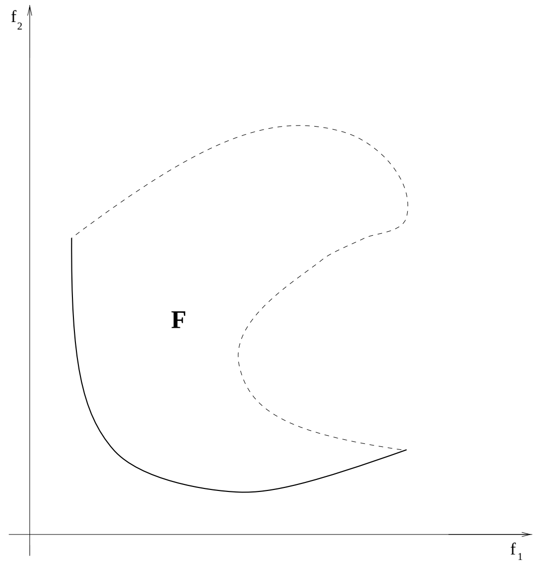

Figure 1: An example of a problem with two objective functions. The Pareto front is marked with a bold line.

approach was to use an extension of the Simple Genetic Algorithm (SGA) that he called the *Vector Evaluated Genetic Algorithm* (VEGA), and that differed of the first only in the way in which selection was performed. This operator was modified so that at each generation a number of sub-populations was generated by performing proportional selection according to each objective function in turn. Thus, for a problem with k objectives, k sub-populations of size N/k each would be generated (assuming a total population size of N). These sub-populations would be shuffled together to obtain a new population of size N, on which the GA would apply the crossover and mutation operators in the usual way. Schaffer realized that the solutions generated by his system were nondominated in a local sense, because their non-dominance was limited to the current population, and while a locally dominated individual is also globally dominated, the converse is not necessarily true [16]. An individual who is not dominated in one generation may become dominated by an individual who emerges in a later generation. Also, he noted a problem that in genetics is known as "speciation" (i.e., we could have the evolution of "species" within the population which excel on different aspects of performance). This problem arises because this technique selects individuals who excel in one dimension of performance, without looking at the other dimensions. The potential danger doing that is that we could have individuals with what Schaffer calls "middling" performance[3] in all dimensions, which could be very useful for compromise solutions, but that will not survive under this selection scheme, since they are not in the extreme for any dimension of performance (i.e., they do not produce the best value for any objective function, but only moderately good values for all of them). Speciation is undesirable because it is opposed to our goal of finding a compromise solution. Schaffer suggested some heuristics to deal with this problem. For example, to use a heuristic selection preference approach for non-dominated individuals in each generation, to protect our "middling" chromosomes. Also, crossbreeding among the "species" could be encouraged by adding some mate selection heuristics instead of using the random mate selection of the traditional GA.

[3] By "middling", Schaffer meant an individual with acceptable performance, perhaps above average, but not outstanding for any of the objective functions.

4.2.1 Applications

Ritzel and Wayland [15] used a variation of VEGA in which they incorporated a parameter to control the selection ratio. In the case of the groundwater pollution containment problem that Ritzel and Wayland solved, there were only two objectives, and the selection ratio was defined as the ratio of the fraction of strings selected on the basis of the first objective (reliability) to the fraction selected via the second objective (cost). Surry et al. [18] proposed an interesting application of VEGA to model constraints in a single-objective optimization problem to avoid the need of a penalty function. Surry et al., however, modified the standard procedure of VEGA and introduced a form of ranking based on the number of constraints violated by a certain solution, and they reported that their approach worked well in the optimization of gas supply networks, since the tendency of VEGA to favor certain solutions can actually be an advantage when handling constraints, because in that case we want to favor precisely any solution that does not violate any constraint over those which do. Cvetković et al. [19] proposed several approaches to overcome VEGA's problems. For example, to wait for a certain amount of generations before shuffling together the population, or avoid shuffling the individuals, and instead copy or migrate a certain amount of individuals from one subpopulation to another. They used these and other traditional multiobjective optimization approaches for preliminary airframe design. Tamaki et al. [20, 21] developed a technique in which at each generation, non-dominated individuals in the current population are kept for the following generation. This approach is really a mixture of Pareto selection and VEGA, because if the number of non-dominated individuals is less that the population size, the remainder of the population in the following generation is filled applying VEGA to the dominated individuals. On the other hand, if the number of the non-dominated individuals exceeds the population size, individuals in the following generation are selected among the non-dominated individuals using VEGA. In a later version of this algorithm, called Pareto Reservation strategy, Tamaki et al. [21] used also fitness sharing among the non-dominated individuals to maintain diversity in the population.

4.2.2 Strengths and weaknesses

Although Schaffer reported some success, and the main strength of this approach is its simplicity, Richardson et al. [22] noted that the shuffling and merging of all the subpopulations corresponds to averaging the fitness components associated with each of the objectives. Since Schaffer used proportional fitness assignment [23], these fitness components were in turn proportional to the objectives themselves [24]. Therefore, the resulting expected fitness corresponded to a linear combination of the objectives where the weights depended on the distribution of the population at each generation as shown by Richardson et al. [22]. The main consequence of this is that when we have a concave trade-off surface certain points in concave regions will not be found through this optimization procedure in which we are using just a linear combination of the objectives, and it has been proved that this is true regardless of the set of weights used [22]. Therefore, the main weakness of this technique is its inability to produce Pareto-optimal solutions in the presence of non-convex search spaces.

4.3 Fonseca and Fleming's MOGA

Fonseca and Fleming [25] have proposed a scheme in which the rank of a certain individual corresponds to the number of chromosomes in the current population by which it is dominated. Consider, for example, an individual x_i at generation t, which is dominated by $p_i^{(t)}$ individuals in the current generation. Its current position in the individuals' rank can be given by [25]:

$$rank(x_i, t) = 1 + p_i^{(t)} \tag{8}$$

All non-dominated individuals are assigned rank 1, while dominated ones are penalized according to the population density of the corresponding region of the trade-off surface.

Fitness assignment is performed in the following way [25]:

1. Sort population according to rank.
2. Assign fitness to individuals by interpolating from the best (rank 1) to the worst (rank $n \leq N$) in the way proposed by Goldberg [23], according to some function, usually linear, but not necessarily.
3. Average the fitnesses of individuals with the same rank, so that all of them will be sampled at the same rate. This procedure keeps the global population fitness constant while maintaining appropriate selective pressure, as defined by the function used.

As Goldberg and Deb [26] point out, this type of blocked fitness assignment is likely to produce a large selection pressure that might produce premature convergence. To avoid that, Fonseca and Fleming used a niche-formation method to distribute the population over the Pareto-optimal region, but instead of performing sharing on the parameter values, they have used sharing on the objective function values [27].

4.3.1 Applications

Chen Tan and Li [28] reported success in the use of MOGA for the multiobjective optimization of ULTIC controllers that satisfy a number of time domain and frequency domain specifications. Also, Chipperfield and Fleming [29] reported success in using MOGA for the design of a multivariable control system for a gas turbine engine. Obayashi [30] used Pareto ranking with phenotypic sharing and *best-N* selection (the best N individuals are selected for the next generation among N parents and N children) for the aerodynamic design of compressor blade shapes. Rodríguez Vázquez et al. [31] extended MOGA to use it in genetic programming, introducing the so-called MOGP (Multiple Objective Genetic Programming). Genetic programming [32] replaces the traditional linear chromosomic representation by a hierarchical tree representation that, by definition, is more powerful, but also requires larger population sizes and specialized operators. MOGP was used for the identification of non-linear model structures, as an alternative that the authors reported to work better (in terms of representation power) than the use of the conventional linear representation of MOGA that they had attempted before [33]. Aherne et al. [34] used MOGA to optimize the selection of parameters for an object recognition scheme called the Pairwise Geometric Histogram paradigm. Todd and Sen [35] used a variant of MOGA for the preplanning of containership layouts (a large scale combinatorial problem). In Todd and Sen's approach, a population of nondominated individuals is kept and updated at each generation, removing individuals that become dominated and duplicates. The traditional genetic operators and sharing are applied only to this population. Niche sizes are computed automatically for each criterion by substracting the maximum value for that criterion from the minimum and dividing it by the population size. Crossover was restricted so that only individuals that were very similar could mate, and because of the permutations encoded, a repair algorithm had to be used afterwards. Finally, a heuristic mutation that basically defined rules to exchange bit positions had to be used to avoid premature convergence of the population.

4.3.2 Strengths and weaknesses

The main strengths of MOGA is that is efficient and relatively easy to implement [36]. Its main weakness is that, as all the other Pareto ranking techniques, its performance is highly dependent on an appropriate selection of the sharing factor. However, it is important to add that Fonseca and Fleming [25] have developed a good methodology to compute such value for their approach.

4.4 Srinivas and Deb's NSGA

The Non-dominated Sorting Genetic Algorithm (NSGA) was proposed by Srinivas and Deb [37], and is based on several layers of classifications of the individuals. Before selection is performed, the population is ranked on the basis of nondomination: all nondominated individuals are classified into one category (with a dummy fitness value, which is proportional to the population size, to provide an equal reproductive potential for these individuals). To maintain the diversity of the population, these classified individuals are shared with their dummy fitness values. Then this group of classified individuals is ignored and another layer of nondominated individuals is considered. The process continues until all individuals in the population are classified. A stochastic remainder proportionate selection was used for this approach. Since individuals in the first front have the maximum fitness value, they always get more copies than the rest of the population. This allows to search for nondominated regions, and results in quick convergence of the population toward such regions. Sharing, by its part, helps to distribute it over this region. The efficiency of NSGA lies in the way in which multiple objectives are reduced to a dummy fitness function using a nondominated sorting procedure. With this approach, any number of objectives can be solved [27], and both maximimization and minimization problems can be handled.

4.4.1 Applications

Périaux et al. [38] used the NSGA to find an optimal distribution of active control elements which minimizes the backscattering of aerodynamic reflectors. Vedarajan et al. [39] used the NSGA for investment portfolio optimization, but interestingly they used binary tournament selection instead of stochastic remainder selection as suggested by Srinivas and Deb [37]. The authors claim that this approach worked well in their examples, although they do not provide any argument for their choice of selection strategy. Tournament selection is expected to introduce a high selection pressure that may dilute the effect of sharing. However, since Vedarajan et al. used fairly large population sizes (above 1000 individuals), the counter-effect of tournament selection may had been absorbed by the extra individuals in the population. Michielssen and Weile [40] used also the NSGA to design an electromagnetic system.

4.4.2 Strengths and weaknesses

The main strength of this technique is that can handle any number of objectives, and that does sharing in the parameter value space instead of the objective value space, which ensures a better distribution of individuals, and allows multiple equivalent solutions exist. Some researchers [36] have reported that its main weakness is that it is more inefficient (both computationally and in terms of quality of the Pareto fronts produced) than MOGA, and more sentitive to the value of the sharing factor σ_{share}. Other authors [41, 42] report that the NSGA performed quite well in terms of "coverage" of the Pareto front (i.e., it spreads in a more uniform way the population over the Pareto front) when applied to the 0/1 knapsack problem, but in their experiments no comparisons with MOGA were provided.

4.5 Horn and Nafpliotis' NPGA

Horn and Nafpliotis [43] proposed a tournament selection scheme based on Pareto dominance. Instead of limiting the comparison to two individuals, a number of other individuals in the population was used to help determine dominance (typically around 10). When both competitors were either dominated or non-dominated (i.e., there was a tie), the result of the tournament was decided through fitness sharing [44]. Population sizes considerably larger than usual with other approaches were used so that the noise of the selection method could be tolerated by the emerging niches in the population [24].

Horn and Nafpliotis [43] arrived at a form of fitness sharing in the objective domain, and suggested the use of a metric combining both the objective and the decision variable domains, leading to what they called *nested sharing*.

4.5.1 Applications

Belegundu et al. [45] used the NPGA for the design of laminated ceramic composites. Poloni and Pediroda [46] used it for the design of a multipoint airfoil that has its minimum drag at two given lift values with a constraint in the maximum allowed pitching moment. A variation of the NPGA was proposed by Quagliarella and Vicini [47]. They introduced the dominance criteria of the problem in the selection mechanism (as in the NPGA), but then selected the individuals to be reproduced to generate the following population using a random walk operator. This obviously produces a locally dominating individual rather than a globally dominating one. Additionally, if more than one non dominated individual is found, then the first one encountered is selected (instead of doing sharing like in the NPGA). At the end of every new generation, the set of Pareto optimal solutions is updated and stored. They used this approach for airfoil design [47].

4.5.2 Strengths and weaknesses

Since this approach does not apply Pareto selection to the entire population, but only to a segment of it at each run, its main strengths are that is very fast and that it produces good non-dominated fronts that can be kept for a large number of generations [36]. However, its main weakness is that besides requiring a sharing factor, this approach also requires a good choice of the size of the tournament to perform well, complicating its appropriate use in practice.

4.6 Target Vector Approaches

Under this name we will consider approaches in which the decision maker has to assign targets or goals that he/she wishes to achieve for each objective. The most popular techniques included here are: Goal Programming [48, 49], Goal Attainment [9, 50] and the min-max approach [51, 52, 53]. These techniques will yield a dominated solution if the goals desired are chosen in the feasible domain, which is a condition that might certainly limit their applicability.

4.6.1 Applications

Wilson & MacLeod [9] used goal-attainment as another of the methods incorporated into their GA to design multiplierless IIR filters. Wienke et al. [54] used goal-programming in combination with a genetic algorithm to optimize simultaneously the intensities of six atomic emission lines of trace elements in alumina powder as a function of spectroscopic excitation conditions. Eric Sandgren [55] also used goal programming coupled with a genetic algorithm to optimize plane trusses and the design of a planar mechanism. Coello and Christiansen applied a weighted min-max approach to the optimization of I-beams [56] and manufacturing problems [53], and to the design of a robot arm [52]. Hajela and Lin [51] applied a weighted min-max approach to several structural optimization problems.

4.6.2 Strengths and weaknesses

The main strength of these methods is their efficiency (computationally speaking) because they do not require any nondominance comparisons. However, their main weakness is the definition of the goals (and probably weights for each objective). These techniques have also been criticized for not being able to deal with non-convex search spaces [5].

5 Theory

Not much theoretical work has been performed in this area, despite the large amount of publications reported in the literature, since most of them deal with either applications or new variations of existing techniques. The most important theoretical work in this area is easily identified:

- Fonseca and Fleming [25, 57, 58] have provided some important (general) concepts on Pareto ranking, nondominance, and ways to determine sharing factors and mating restriction parameters.
- Horn [59, 60] and Horn and Nafpliotis [61] have provided important guidelines to choose appropriate values for the sharing factor.
- Rudolph [62] and Van Veldhuizen and Lamont [63] have provided some theoretical analysis of convergence towards the Pareto set in an attempt to define the limits of GA-based search in this domain.

Obviously, a lot of work remains to be done regarding theory of EMOO techniques. First, it would be desirable to perform a rigorous and detailed analysis of some of the most common EMOO approaches. Apparently, Fonseca and Fleming's MOGA is a good candidate for this sort of detailed mathematical analysis [5]. Another important theoretical aspect that deserves attention is the analysis of the size of the Pareto front with respect to the objectives. Although some researchers [57, 59] have implied that the size of the Pareto front grows with the number of objectives, some recent work by Van Veldhuizen and Lamont [5, 63] indicate that the number of points in the Pareto front really depends more on the front's shape than on the number of objectives.

It is also required more work on niching and fitness sharing so that more accurate sharing factors can be easily defined. Finally, there is a need for detailed studies of the different aspects involved in the parallelization of EMOO approaches (e.g., load balancing, impact on Pareto convergence, performance issues, etc.), including new algorithms that are more suitable for parallelization than those currently in use.

6 Test Functions

A very important aspect of this research area that has been generally disregarded in the technical literature is the use of appropriate test functions. For some reason, many researchers have tested their approaches only with the two classic test functions provided by Schaffer in his seminal work on EMOO [16]. These functions are not only very simple (they have only two objectives), but are also unconstrained and do not show any of the most important aspects that would be interesting to analyze using an EMOO approach (e.g., concavity or discontinuity of the Pareto front).

In this direction, Deb [64, 65] has recently proposed ways to create controllable test problems for evolutionary multiobjective optimization techniques using single-objective optimization problems as a basis. This is an interesting proposal that could allow to transform deceptive and massively multimodal problems into very difficult multiobjective optimization problems. Van Veldhuizen and Lamont [5] have also proposed some guidelines to design a test function suite for evolutionary multiobjective optimization techniques, and have included in a technical report some sample test problems (mainly combinatorial optimization problems) [5].

Using benchmark problems such as those proposed by Deb [64, 65] and Van Veldhuizen & Lamont [5], it should be possible to perform detailed studies of performance of different GAs (assuming certain quality measures) which are also notoriously lacking in the technical literature (see Van Veldhuizen and Lamont [5] for a detailed account of the few existing comparative studies of EMOO techniques).

7 Metrics

It is very important to define good metrics to measure the effectiveness of an EMOO technique. The main proposals so far are the following:

- To enumerate the entire intrinsic search space using parallel processing techniques [5] so that we can know the true Pareto front within the representation used. If we know the true Pareto front, we can compare our results against it, and devise different metrics for estimating how well is our GA performing (e.g., Van Veldhuizen and Lamont [63] proposed the so-called *generational distance*, which represents how far is a certain solution from the true Pareto front).

  This approach might work with relatively short binary strings (Van Veldhuizen [5] reports success with strings ≤ 26 bits), but might not be suitable when using alphabets of higher cardinality (e.g., real-coded GAs) or longer binary strings.
- Srinivas and Deb [27] proposed to measure the "spread" of points along the Pareto front using a statistical metric such as the chi-square distribution. This metric also assumes knowledge of the true Pareto front, and emphasizes the good distribution of points (determined through a set of niches) rather than a direct comparison between our Pareto front and the true Pareto front.
- Zitzler and Thiele [42] proposed two measures: the first concerns the size of the objective value space which is covered by a set of nondominated solutions and the second compares directly two sets of nondominated solutions, using as a metric the fraction of the Pareto front covered by each of them.

All these metrics are interesting proposals but there are almost no comparative studies of techniques that substantiate their suitability in general test problems, which implies that more work in this area is required.

8 Future Research Paths

As has been indicated before in some of the sections of this paper, a lot of work remains to be done in this area. We will briefly describe here some of the future research paths that have not been mentioned yet and some of the related work done so far (if any):

- Dynamics of a population: As Deb indicates [64], it would be very useful to understand the dynamics of the population of a GA over different generations when applied to multiobjective optimization problems. If we knew how is the population behaving and what issues are making it difficult to keep nondominated solutions, we could devise techniques in which the progress towards the global Pareto front could be considerably faster than with the current approaches.
- Stopping criteria: It is also important to define stopping criteria for a GA-based multiobjective optimization technique, because it is not obvious to know when the population has reached a point from which no further improvement can be reached (i.e., how do we know that the global Pareto front has been found?). Currently, the main approaches used to stop this kind of GA have been to either use a fixed number of generations, or to monitor the population at certain intervals and interpret visually the results to determine when to halt the evolution process.
- Real-world applications: With no doubt, the number of applications of evolutionary multiobjective optimization techniques to real-world problems will increase over the years, and a probable trend in research could be to reformulate many problems that are currently considered as if they only had one objective (e.g., constraint-handling in single-objective optimization [66]). This will constitute a more realistic approach to the solution of problems that frequently arise in areas such as engineering, because they are normally reduced to a single objective and the remaining objectives are treated as constraints instead of handling all (conflicting) objectives simultaneously.

9 Conclusions

This paper has attempted to provide a general view of the main work that has been done on evolutionary multiobjective optimization, discussing the most popular techniques currently in use, some of their applications, their advantages and disadvantages, and some of the most important problems that remain to be solved and the related work in process regarding their (possible) solution.

Acknowledgments

The author acknowledges support from CONACyT through project number I-29870 A.

Bibliography

[1] R. S. Rosenberg. *Simulation of genetic populations with biochemical properties*. PhD thesis, University of Michigan, Ann Harbor, Michigan, 1967.

[2] Andrzej Osyczka. Multicriteria optimization for engineering design. In John S. Gero, editor, *Design Optimization*, pages 193–227. Academic Press, 1985.

[3] Vilfredo Pareto. *Cours D'Economie Politique*, volume I and II. F. Rouge, Lausanne, 1896.

[4] Carlos A. Coello Coello. A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. *Knowledge and Information Systems. An International Journal*, 1999. (Accepted for publication).

[5] David A. Van Veldhuizen and Gary B. Lamont. Multiobjective Evolutionary Algorithm Research: A History and Analysis. Technical Report TR-98-03, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, 1998.

[6] Gilbert Syswerda and Jeff Palmucci. The Application of Genetic Algorithms to Resource Scheduling. In Richard K. Belew and Lashon B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 502–508, San Mateo, California, 1991. Morgan Kaufmann.

[7] W. Jakob, M. Gorges-Schleuter, and C. Blume. Application of genetic algorithms to task planning and learning. In R. Männer and B. Manderick, editors, *Parallel Problem Solving from Nature, 2nd Workshop*, Lecture Notes in Computer Science, pages 291–300, Amsterdam, 1992. North-Holland Publishing Company.

[8] Gareth Jones, Robert D. Brown, David E. Clark, Peter Willett, and Robert C. Glen. Searching Databases of Two-Dimensional and Three-Dimensional Chemical Structures using Genetic Algorithms. In Stephanie Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 597–602, San Mateo, California, 1993. Morgan Kaufmann.

[9] P. B. Wilson and M. D. Macleod. Low implementation cost IIR digital filter design using genetic algorithms. In *IEE/IEEE Workshop on Natural Algorithms in Signal Processing*, pages 4/1–4/8, Chelmsford, U.K., 1993.

[10] Xiaojian Liu, D. W. Begg, and R. J. Fishwick. Genetic approach to optimal topology/controller design of adaptive structures. *International Journal for Numerical Methods in Engineering*, 41:815–830, 1998.

[11] Xiaofeng Yang and Mitsuo Gen. Evolution program for bicriteria transportation problem. In M. Gen and T. Kobayashi, editors, *Proceedings of the 16th International Conference on Computers and Industrial Engineering*, pages 451–454, Ashikaga, Japan, 1994. Pergamon Press.

[12] M. Gen, K. Ida, and Y. Li. Solving bicriteria solid transportation problem with fuzzy numbers by genetic algorithm. *International Journal of Computers and Industrial Engineering*, 29:537–543, 1995.

[13] Mitsuo Gen and Runwei Cheng. *Genetic Algorithms and Engineering Design*. John Wiley and Sons, Inc., New York, 1997.

[14] H. W. Kuhn and A. W. Tucker. Nonlinear programming. In J. Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, California, 1951. University of California Press.

[15] Brian J. Ritzel, J. Wayland Eheart, and S. Ranjithan. Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. *Water Resources Research*, 30(5):1589–1603, may 1994.

[16] J. David Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, pages 93–100. Lawrence Erlbaum, 1985.

[17] J. J. Grefenstette. GENESIS: A system for using genetic search procedures. In *Proceedings of the 1984 Conference on Intelligent Systems and Machines*, pages 161–165, 1984.

[18] Patrick D. Surry, Nicholas J. Radcliffe, and Ian D. Boyd. A Multi-Objective Approach to Constrained Optimisation of Gas Supply Networks : The COMOGA Method. In Terence C. Fogarty, editor, *Evolutionary Computing. AISB Workshop. Selected Papers*, Lecture Notes in Computer Science, pages 166–180. Springer-Verlag, Sheffield, U.K., 1995.

[19] Dragan Cvetković, Ian Parmee, and Eric Webb. Multi-objective Optimisation and Preliminary Airframe Design. In Ian Parmee, editor, *The Integration of Evolutionary and Adaptive Computing Technologies with Product/System Design and Realisation*, pages 255–267, Plymouth, United Kingdom, April 1998. Plymouth Engineering Design Centre, Springer-Verlag.

[20] Hisashi Tamaki, M. Mori, M. Araki, Y. Mishima, and H. Ogai. Multi-Criteria Optimization by Genetic Algorithms : A Case of Scheduling in Hot Rolling Process. In *Proceedings of the 3rd Conference of the Association of Asian-Pacific Operational Research Societies within IFORS (APORS'94)*, pages 374–381. World Scientific, 1995.

[21] Hisashi Tamaki, Hajime Kita, and Shigenobu Kobayashi. Multi-Objective Optimization by Genetic Algorithms : A Review. In Toshio Fukuda and Takeshi Furuhashi, editors, *Proceedings of the 1996*

International Conference on Evolutionary Computation (ICEC'96), pages 517–522, Nagoya, Japan, 1996. IEEE.

[22] Jon T. Richardson, Mark R. Palmer, Gunar Liepins, and Mike Hilliard. Some guidelines for genetic algorithms with penalty functions. In J. David Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, pages 191–197, George Mason University, 1989. Morgan Kaufmann Publishers.

[23] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.

[24] Carlos M. Fonseca and Peter J. Fleming. An overview of evolutionary algorithms in multiobjective optimization. Technical report, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, U. K., 1994.

[25] Carlos M. Fonseca and Peter J. Fleming. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In Stephanie Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 416–423, San Mateo, California, 1993. University of Illinois at Urbana-Champaign, Morgan Kauffman Publishers.

[26] David E. Goldberg and Kalyanmoy Deb. A comparison of selection schemes used in genetic algorithms. In G.J. E. Rawlins, editor, *Foundations of Genetic Algorithms*, pages 69–93. Morgan Kaufmann, San Mateo, California, 1991.

[27] N. Srinivas and Kalyanmoy Deb. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3):221–248, fall 1994.

[28] Kay Chen Tan and Yun Li. Multi-Objective Genetic Algorithm Based Time and Frequency Domain Design Unification of Linear Control Systems. Technical Report CSC-97007, Department of Electronics and Electrical Engineering, University of Glasglow, Glasglow, Scotland, 1997.

[29] A. J. Chipperfield and P. J. Fleming. Gas Turbine Engine Controller Design using Multiobjective Genetic Algorithms. In A. M. S. Zalzala, editor, *Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems : Innovations and Applications, GALESIA'95*, pages 214–219, Halifax Hall, University of Sheffield, UK, September 1995. IEEE.

[30] Shigeru Obayashi. Pareto Genetic Algorithm for Aerodynamic Design using the Navier-Stokes Equations. In D. Quagliarella, J. Périaux, C. Poloni, and G. Winter, editors, *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science. Recent Advances and Industrial Applications*, chapter 12, pages 245–266. John Wiley and Sons, West Sussex, England, 1997.

[31] Katya Rodríguez-Vázquez, Carlos M. Fonseca, and Peter J. Fleming. Multiobjective Genetic Programming : A Nonlinear System Identification Application. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1997 Conference*, pages 207–212, Stanford University, California, July 1997. Stanford Bookstore.

[32] John R. Koza. *Genetic Programming. On the Programming of Computers by Means of Natural Selection*. The MIT Press, 1992.

[33] Carlos M. Fonseca and Peter J. Fleming. Nonlinear System Identification with Multiobjective Genetic Algorithms. In *Proceedings of the 13th World Congress of the International Federation of Automatic Control*, pages 187–192, San Francisco, California, 1996. Pergamon Press.

[34] Frank J. Aherne, Neil A. Thacker, and Peter I. Rockett. Optimal Pairwise Geometric Histograms. In Adrian F. Clark, editor, *Electronic Proceedings of the Eighth British Machine Vision Conference, BMVC97*, University of Essex, United Kingdom, September 1997.

[35] David S. Todd and Pratyush Sen. A Multiple Criteria Genetic Algorithm for Containership Loading. In Thomas Bäck, editor, *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 674–681, San Mateo, California, July 1997. Michigan State University, Morgan Kaufmann Publishers.

[36] Carlos Artemio Coello Coello. *An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design*. PhD thesis, Department of Computer Science, Tulane University, New Orleans, LA, apr 1996.

[37] N. Srinivas and Kalyanmoy Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. Technical report, Department of Mechanical Engineering, Indian Institute of Technology, Kanput, India, 1993.

[38] Jacques Périaux, Mourad Sefrioui, and Bertrand Mantel. GA Multiple Objective Optimization Strategies for Electromagnetic Backscattering. In D. Quagliarella, J. Périaux, C. Poloni, and G. Winter, editors, *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science. Recent Advances and Industrial Applications*, chapter 11, pages 225–243. John Wiley and Sons, West Sussex, England, 1997.

[39] Ganesh Vedarajan, Louis Chi Chan, and David E. Goldberg. Investment Portfolio Optimization using Genetic

Algorithms. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1997 Conference*, pages 255–263, Stanford University, California, July 1997. Stanford Bookstore.

[40] E. Michielssen and D. S. Weile. Electromagnetic System Design using Genetic Algorithms. In *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science*, pages 267–288. John Wiley and Sons, England, 1995.

[41] Eckart Zitzler and Lothar Thiele. Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Study. In A. E. Eiben, editor, *Parallel Problem Solving from Nature V*, pages 292–301, Amsterdam, September 1998. Springer-Verlag.

[42] Eckart Zitzler and Lothar Thiele. An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, May 1998.

[43] Jeffrey Horn and Nicholas Nafpliotis. Multiobjective Optimization using the Niched Pareto Genetic Algorithm. Technical Report IlliGAl Report 93005, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA, 1993.

[44] David E. Goldberg and J. Richardson. Genetic algorithm with sharing for multimodal function optimization. In J. J. Grefenstette, editor, *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, pages 41–49. Lawrence Erlbaum, 1987.

[45] A. D. Belegundu, D. V. Murthy, R. R. Salagame, and E. W. Constants. Multiobjective Optimization of Laminated Ceramic Composites Using Genetic Algorithms. In *Fifth AIAA/USAF/NASA Symposium on Multidisciplinary Analysis and Optimization*, pages 1015–1022, Panama City, Florida, 1994. AIAA. Paper 84-4363-CP.

[46] Carlo Poloni and Valentino Pediroda. GA coupled with computationally expensive simulations: tools to improve efficiency. In D. Quagliarella, J. Périaux, C. Poloni, and G. Winter, editors, *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science. Recent Advances and Industrial Applications*, chapter 13, pages 267–288. John Wiley and Sons, West Sussex, England, 1997.

[47] Domenico Quagliarella and Alessandro Vicini. Coupling Genetic Algorithms and Gradient Based Optimization Techniques. In D. Quagliarella, J. Périaux, C. Poloni, and G. Winter, editors, *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science. Recent Advances and Industrial Applications*, chapter 14, pages 289–309. John Wiley and Sons, West Sussex, England, 1997.

[48] A. Charnes and W. W. Cooper. *Management Models and Industrial Applications of Linear Programming*, volume 1. John Wiley, New York, 1961.

[49] Y. Ijiri. *Management Goals and Accounting for Control*. North-Holland, Amsterdan, 1965.

[50] Y. L. Chen and C. C. Liu. Multiobjective VAR planning using the goal-attainment method. *IEE Proceedings on Generation, Transmission and Distribution*, 141(3):227–232, may 1994.

[51] P. Hajela and C. Y. Lin. Genetic search strategies in multicriterion optimal design. *Structural Optimization*, 4:99–107, 1992.

[52] Carlos A. Coello Coello, Alan D. Christiansen, and Arturo Hernández Aguirre. Using a new GA-based multiobjective optimization technique for the design of robot arms. *Robotica*, 16:401–414, 1998.

[53] Carlos A. Coello Coello and Alan D. Christiansen. Two new GA-based methods for multiobjective optimization. *Civil Engineering Systems*, 1998. (In Press).

[54] P. B. Wienke, C. Lucasius, and G. Kateman. Multicriteria target optimization of analytical procedures using a genetic algorithm. *Analytical Chimica Acta*, 265(2):211–225, 1992.

[55] Eric Sandgren. Multicriteria design optimization by goal programming. In Hojjat Adeli, editor, *Advances in Design Optimization*, pages 225–265. Chapman & Hall, London, 1994.

[56] Carlos A. Coello Coello and Alan D. Christiansen. MOSES : A Multiobjective Optimization Tool for Engineering Design. *Engineering Optimization*, 1998. (Accepted for publication).

[57] Carlos M. Fonseca and Peter J. Fleming. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3(1):1–16, Spring 1995.

[58] Carlos M. Fonseca and Peter J. Fleming. On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature—PPSN IV*, Lecture Notes in Computer Science, pages 584–593. Springer-Verlag, Berlin, Germany, September 1996.

[59] Jeffrey Horn. Multicriterion Decision making. In Thomas Bäck, David Fogel, and Zbigniew Michalewicz, editors, *Handbook of Evolutionary Computation*, volume 1, pages F1.9:1 – F1.9:15. IOP Publishing Ltd. and Oxford University Press, 1997.

[60] Jeffrey Horn. *The Nature of Niching: Genetic Algorithms and the Evolution of Optimal, Cooperative Populations*. PhD thesis, University of Illinois at Urbana Champaign, Urbana, Illinois, 1997.

[61] Jeffrey Horn, Nicholas Nafpliotis, and David E. Goldberg. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, pages 82–87, Piscataway, New Jersey, June 1994. IEEE Service Center.

[62] Günter Rudolph. On a Multi-Objective Evolutionary Algorithm and Its Convergence to the Pareto Set. In *Proceedings of the 5th IEEE Conference on Evolutionary Computation*, pages 511–516, Piscataway, New Jersey, 1998. IEEE Press.

[63] David A. Van Veldhuizen and Gary B. Lamont. Evolutionary Computation and Convergence to a Pareto Front. In John R. Koza, editor, *Late Breaking Papers at the Genetic Programming 1998 Conference*, pages 221–228, Stanford University, California, July 1998. Stanford University Bookstore.

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

[65] Kalyanmoy Deb. Evolutionary Algorithms for Multi-Criterion Optimization in Engineering Design. In *Proceedings of Evolutionary Algorithms in Engineering and Computer Science (EUROGEN'99)*, 1999.

[66] Carlos A. Coello Coello. Treating Constraints as Objectives for Single-Objective Evolutionary Optimization. *Engineering Optimization*, 32, 1999. (Accepted for publication).