F1.12: Multicriteria Decision Making and Evolutionary Computation

Jeffrey Horn *
Department of Mathematics and Computer Science
161 West Science Building
Northern Michigan University
Marquette, MI 49855
(e-mail: jeffhorn@uiuc.edu)

Submitted to *The Handbook of Evolutionary Computation*, July 1996
(also to be made available as IlliGAL Report No. 9600X, University of Illinois)

July 29, 1996

Abstract

Applying evolutionary computation (EC) to multicriteria decision making addresses two difficult problems: (1) searching intractably large and complex spaces and (2) deciding among multiple objectives. Both of these problems are open areas of research, but relatively little work has been done on the COMBINED problem of searching large spaces to meet multiple objectives. While multicriteria decision analysis usually assumes a small number of alternative solutions to choose from, or an "easy" (e.g., linear) space to search, research on robust search methods generally assumes some way of aggregating multiple objectives into a single figure of merit. This traditional separation of search and multicriteria decisions allows for two straightforward hybrid strategies: (1) make multicriteria decisions FIRST, to aggregate objectives, then apply EC search to optimize the resulting figure of merit, or (2) conduct multiple EC searches FIRST using different aggregations of the objectives in order to obtain a range of alternative solutions, then make a multicriteria decision to choose among the reduced set of solutions. Over the years a number of studies have successfully used one or the other of these two simple hybrid approaches. Recently, however, many studies have implemented Pareto-based EC search to sample the entire Pareto-optimal set of non-dominated solutions. And a few researchers have suggested ways of INTEGRATING multicriteria decision making and EC search, by iteratively using EC search to sample the tradeoff surface while using multicriteria decision making to successively narrow the search. Although all these approaches have received only limited testing and analysis, there are few comparable alternatives to multicriteria EC search (for searching intractably large spaces to meet multiple criteria).

1 INTRODUCTION

One could argue that real world problems are, in general, multicriteria. That is, the problems involve multiple objectives[1] to be met or optimized, with the objectives often conflicting. The application of evolutionary search to multicriteria problems seems a logical next step for the evolutionary computation (EC) approaches that have been successful on hard single-criterion problems. And indeed, quite a few EC approaches have found very satisfactory "tradeoff solutions" in multiobjective problems. But EC search can be, and has been, applied to multiobjective problems in a number of very different ways. It is far from

*The author acknowledges support provided by NASA under contract number NGT-50873, the US Army under Contract DASG60-90-C-0153 and by the National Science Foundation under Grant ECS-9022007, while at the University of Illinois at Urbana-Champaign.

[1]The terms *objective, criterion,* and *attribute* are sometimes subtly distinguished in the literature, but here they are used interchangeably to mean one of a set of goals to be achieved (e.g., cost, to be reduced).

clear which, if any, approaches are superior for general classes of multiobjective problems. At this early point in the development of multicriteria ECs, it would be a good idea to try more than one approach on any given problem.

2 DESCRIPTION OF THE MULTICRITERIA APPLICATION DOMAIN

Multicriteria problems in general involve two "quasi-separable" types of problem difficulty: **search** and **multicriteria decision making**. As with single criterion problems, the space to be searched can be too large to be enumerated, and too complex (e.g., multimodal, nonlinear) to be solved by linear programming or local or gradient search. In addition to search space complexity, the multiple objectives to be achieved may be conflicting, so that difficult tradeoffs must be made by a rational human decision maker (DM) when ranking solutions. (Indeed, as Goldberg (1989) points out, if our multiple objectives never conflict over the set of feasible solutions, then we do not have any difficulty with the multiple objectives. The search space is then completely (totally) ordered, not just partially ordered, and any reasonable aggregation of the multiple objectives into a single objective will maintain this ordering.)

Traditionally these two aspects of the overall problem, search and multicriteria decision making, are treated separately, and often one or the other is assumed away. Most approaches to searching intractably large spaces (e.g., EC, simulated annealing (SA), tabu search, stochastic hillclimbing) assume a single objective to be optimized. At the same time, the extensive literature on multiobjective optimization generally assumes a small, enumerable search space, so that the multicriteria decision, not search, is the focus of analysis. EC is in a unique position to address BOTH search and multicriteria decisions because of its ability to search partially ordered spaces for multiple alternative tradeoffs. Here we assume that both difficulties are necessarily present, or we would not have a *multicriteria search problem*, suited for multicriteria EC optimization.

2.1 Practical Applications

Multicriteria problems are common. For example, imagine a manufacturing design problem, involving a number of decision variables (e.g., materials, manufacturing processes), and two criteria: manufacturing **cost** and product **quality**. Cost and quality are often conflicting: using more durable materials in the product increases its useful lifetime but increases cost as well. This conflict gives rise to the multicriteria decision problem: what is the optimal tradeoff of cost versus quality? Other possible objectives include lowering *risk* or *uncertainty*, and reducing the number of constraint violations (Richardson, *et. al.*, 1989; Liepins, *et. al.*, 1990). Another common source of multiple conflicting objectives is the case of multiple decision makers with different preferences, and hence different orderings of the alternatives. Even if each DM could aggregate his or her different criteria into a single ranking of all of the possible alternatives, satisfying ALL of the DMs is a multicriteria problem, with each DM's ranking (rating or ordering) being treated as a separate criterion.

2.2 Search Space

More formally, we assume a multicriteria problem is characterized by a vector of d decision variables and k criteria. The vector of decision variables can be denoted: $\vec{X} = (x_0, x_1, x_2, ..., x_{d-1})$, just as with any single objective optimization problem. But in the multiobjective case the evaluation function F is vector-valued: $F : \vec{X} \rightarrow \vec{A}$, where $\vec{A} = (a_0, a_2, ..., a_{k-1})$ for the k attributes. Thus $F(\vec{X}) = (f_0(\vec{X}), f_1(\vec{X}), ..., f_{k-1}(\vec{X}))$, where $f_i(\vec{X})$ denotes a function mapping the decision variable vector to the range of the single attribute a_i.

Search and multicriteria decisions are not independent tasks. Making some multicriteria choices before search can reduce the size of the search space, while search before decision making can eliminate the vast number of inferior (dominated) solutions and focus decision making on a few clear alternatives. Thus the integration of search and multicriteria decision making is a key issue in EC approaches to this application domain, and the type and degree of such integration distinguishes three major categories of multiobjective EC algorithms, below.

3 EVOLUTIONARY COMPUTATION APPROACHES

For all of these approaches, the issues of solution representation (i.e., chromosomal encoding), and genetic variation (i.e., the recombination and mutation operators), are the same as for traditional, single criterion EC applications. There are no special considerations for choosing the encoding or designing the crossover and mutation operators (with the exception of possible mating restrictions for Pareto-based approaches). The major difference in the multicriteria case is in the objective function. The different (i.e., vector-valued) objective function affects the design of the fitness function and the selection operator, thus these are the EC components we focus on below.

Here we choose to classify approaches according to how they handle the two problems of search and multicriteria decisions. At the highest level, there are three general orderings for conducting search and making multicriteria decisions:

1. Make Multicriteria Decisions **before** Search (Decide $\Rightarrow$ Search),
2. Search **before** Making Multicriteria Decisions (Search $\Rightarrow$ Decide), and
3. Integrate Search and Multicriteria Decision Making (Decide $\Leftrightarrow$ Search).

3.1 Multicriteria Decisions BEFORE Search: Aggregation

By far the most common method of handling multiple criteria, with or without EC search, is to *aggregate* the multiple objectives into a single objective, which is then used to totally order the solutions. Aggregative methods can be further divided into the *scalar-aggregative* and the *order-aggregative* (*non-scalar*) approaches.

3.1.1 Scalar-aggregative

The most common aggregative methods combine the various objectives into a single scalar valued *utility function*, $U(\vec{A})$, where $U : \Re^k \rightarrow \Re$, reflecting the multicriteria tradeoff preferences of a particular DM. The composite function $U \circ F$ can then be used as the fitness function for EC. A scalar fitness function is required for certain types of selection methods, such as fitness-proportionate selection (e.g., roulette wheel, stochastic remainder).

The simplest example of a scalar aggregation is a *linear combination* (a.k.a, *weighted sum*), such as $U(\vec{A}) = w_0 a_o + w_1 a_1 + ... w_{k-1} a_{k-1}$, where the w_i are constant coefficients (i.e., weights). The DM sets the weights to try to account for his or her relative ratings of the attributes. For example, Bhanu and Lee (1994, ch.s 4, 8) sum five measures of image segmentation quality into a single objective using equal weights, while Vemuri and Cedeño (1995) sum the k criteria rankings, rather than the attribute values themselves. One drawback of the linear combination approach is that it can only account for linear relationships among the criteria. It can be generalized to handle nonlinearities by introducing nonlinear terms, such as exponentiating critical attributes, or multiplying together pairs of highly dependent attributes. One can introduce such nonlinear terms in an ad-hoc manner, guided only by intuition and trial-and-error, but we restrict our discussion to more systematic and generalizable methods below.

A very common *nonlinear* scalar aggregation is the **constraint** approach. Constraints can handle nonlinearities that arise when a DM has certain *thresholds* for criteria, that is, maximum or minimum values. For example, a DM might be willing to sacrifice quality to save money, but only down to a certain level. Typically, when a solution fails to meet a constraint, its utility is given a large *penalty*, such as having a large fixed value subtracted or divided into the total score, (e.g., Simpson, Dandy, & Murphy, 1994; Savic, & Walters, 1995). Richardson, *et. al.* (1989) give guidelines for using penalty functions with GAs, while Stanley and Mudge (1995) discuss turning constraints "back into" objectives.

A recent development in the decision analysis community handles multiplicative nonlinearities: **multiattribute utility analysis (MAUA)** (Keeney & Raiffa, 1976; de Neufville, 1990; Horn and Nafpliotis, 1993). Under MAUA, separate utility functions for each attribute, $u_i(a_i)$ are determined for a particular DM in a systematic way, and incorporate attitude toward uncertainty (in each single attribute). These individual utility functions are then combined by multiplication (rather than addition). Through a series of lottery-based questions, the DM's pair-wise tradeoffs (between pairs of attributes) are estimated, and incorporated into the coefficients (weights) for the multiplicative terms.

A similar multiplicative aggregation is used by Wallace, Jakiela, and Flowers (1994). They first determine a DM's *probability of acceptance* function for each criterion, to take into account nonlinear attitudes toward individual criteria (e.g., thresholds). The acceptance probability functions are then multiplied together, giving the overall probability of acceptance, and the logarithm taken (to reduce selection pressure).

Another nonlinear scalar aggregative method is the **distance-to-target** approach. A target attribute vector $\vec{T}$ is chosen as an ideal solution to shoot for. Solutions are evaluated by simply measuring their distance to this theoretical goal in criteria space (e.g., Wienke, Lucasius, & Kateman, 1992). Choosing the target vector and form of the metric both involve multicriteria decisions by the DM. In the case of the metric, the scaling of the attributes greatly affects the relative distances to the goal, while the actual formula for the metric can also change the ordering of solutions. Consider the general class of *Holder* metrics,

$$h_p(\vec{A}, \vec{B}) = \left(\sum_{i=0}^{k-1} |a_i - b_i|^p \right)^{1/p}, p \geq 1. \tag{1}$$

For example, $p = 1$, which is known as the *metropolitan* or "city-block" metric, gives a linear combination of attributes, while the more common $p = 2$ Euclidean distance introduces nonlinearities. In general, increasing the order p of the Holder norm increases the degree of nonlinear interaction among attributes.

More sophisticated refinements of the simple target distance measure include the *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) approach, which seeks to minimize the distance to a "positive ideal solution" while simultaneously maximizing the distance from a "negative ideal solution". (Thus TOPSIS attempts to reduce a k-criteria problem to a $k = 2$, bicriteria one.) Hwang, Lai, and Lui (1993) use TOPSIS to aggregate multiple objectives for GA optimization.

A significant variant of the distance-to-target approach is the *minimax*, or *MinMax*, formulation (Osyczka, 1984; Srinivas, & Deb, 1995). Minimax seeks to minimize the maximum "criteria distance" to the target solution $\vec{T}$. Choosing the maximum of the k criteria distances is equivalent to using the *maximum Holder metric*, which is obtained as $p \to \infty$ in Equation 1 above: $h_\infty(\vec{A}, \vec{T}) = Max(|a_0 - t_0|, |a_1 - t_1|, ..., |a_{k-1} - t_{k-1}|)$. Minimizing this distance becomes the single objective. By differentially scaling the individual criteria differences in the minimax calculation, we obtain *Tschebycheff's weighting method* (Steuer, 1986), which, unlike linear aggregations, can be used to sample concave portions of the Pareto-optimal frontier (Cieniawski, 1993) (see below).

3.1.2 Order-aggregative

Not all aggregations are scalar. For example, the *lexicographic approach* gives a total ordering of all solutions (thus they can be ranked from best to worst), without assigning scalar values. The approach requires the DM to order the criteria. Solutions are then ranked by considering each attribute in order. As in a dictionary, lexicographic ordering first orders items by their most important attribute. If this results in a tie, then the second attribute is considered, and so on. Fourman (1985) uses a lexicographic ordering when comparing individuals under tournament selection in a GA. Non-scalar, order-aggregative methods (e.g., lexicographic ordering, voting schemes) impose a total ordering on the space of solutions, but do not

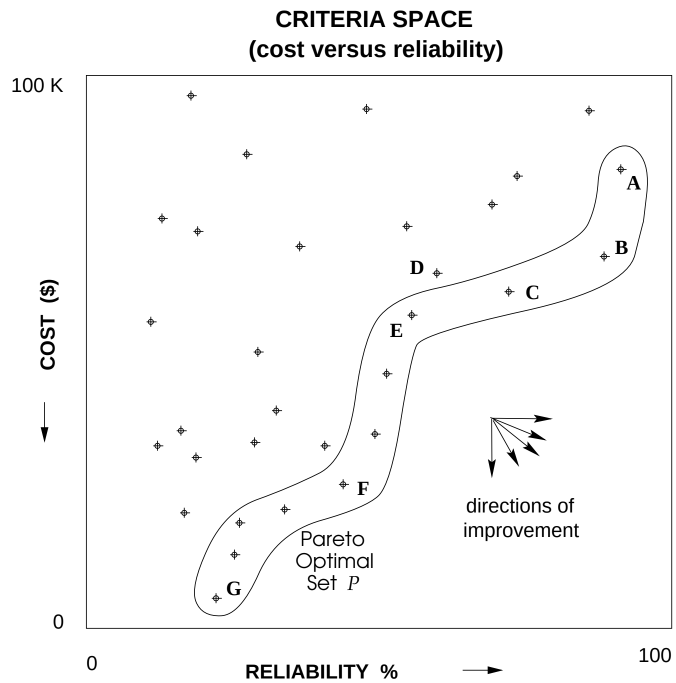

Figure 1: Criteria space in an example two-criteria problem.

provide any meaningful scalar evaluation useful for a fitness function. Therefore such methods are best suited for rank-based selection, including tournament selection.

3.2 Search BEFORE Multicriteria Decisions: Seeking the Pareto Frontier

The aggregative approaches are open to the criticism of being overly simplistic. Is it possible to combine the conflicting objectives into a single preference system, prior to search? Or are some criteria truly non-commensurate? Recognizing the difficulty, perhaps the impossibility, of making all of the multicriteria decisions up front, many users and researchers have chosen to first apply search to find a set of "best alternatives". Multicriteria decision making methods can then be applied to the reduced set of solutions.

Vilfredo Pareto (1896) recognized that even without making any multicriteria decisions, the solution space is already *partially ordered*. Simply stated, the *Pareto criterion* for one solution to be superior to another is for it to be at least as good in all attributes, and superior in at least one. More formally, given k attributes, all of which are to be maximized, a solution **A**, with attribute values $(a_0, a_1, a_2, ..., a_{k-1})$, and a solution **B** with attribute values $(b_0, b_1, b_2, ..., b_{k-1})$, we say that **A** *dominates* **B** if and only if $\forall i : a_i \geq b_i$, and $\exists j : a_j > b_j$. The binary relation of dominance partially orders the space of alternatives. Some pairs of solutions will be *incomparable*, in that neither dominates the other (since one solution might be better than the other in some attributes, and worse in others). Clearly this partial ordering will be agreed to by all rational DMs. Therefore, all dominated solutions can be eliminated from consideration before the multicriteria decisions are made. In particular, the set of solutions not dominated by any solution in the entire space is desirable in that such a set MUST contain all of the possible optimal solutions according to ANY rational DM's multicriteria decisions. This *non-dominated set* is known by many names: the *Pareto optimal set*, the *admissible set*, the *efficient points*, *non-dominated frontier*, and the *Pareto front*, for example.

The terms "front" and "frontier" come from the geometric depiction of the "criteria space" (or "attribute space"). Figure 1 depicts a $k = 2$ criteria space. Here the chosen criteria are cost (to be minimized) and reliability (to be maximized). A population of individuals (candidate solutions) is plotted using each individual's evaluated criteria vector as a coordinate. Note that in Figure 1 individual **C** dominates **D**, but does not dominate **E**. The set of all individuals not dominated by any member of the population (i.e., P) is circled.

The Pareto optimal set P is desirable as input to the multicriteria decision making process for several reasons. (1) Knowledge of the nature of P might simplify the multicriteria decision. For example, P might be singular ($|P| = 1$), with one solution dominating the rest. Or P might at least be small enough to allow a DM, or a team of DMs, to consider all choices at once, in detail. Even if P is large, one solution might stand out, such as an extremum (e.g., **A** or **G** in Figure 1), or perhaps a "knee" of the front (e.g., **B** or **F** in Figure 1), at which large sacrifices in one attribute yield only small improvements in the other(s). (2) P is "DM-independent". If a DM (or his or her preferences) change, the Pareto search need not be performed again. (3) Interpolating a smooth curve through the samples (P) of the front, although potentially misleading, can give some idea of how the attributes interact, and so focus subsequent search on poorly sampled but promising directions (Fonseca & Fleming, 1993a,b; Horn & Nafpliotis, 1993).

All of the approaches described in this section seek the Pareto optimal set (although some authors do not mention Pareto optimality and instead talk of simply finding multiple good tradeoffs). On the notation "P": All of the methods below attempt to evolve a population toward the actual Pareto frontier, call this P_{actual}. Any given population (e.g., generation) has a nondominated subset of individuals, P_{online}. The hope is that by the end of the run, $P_{online} = P_{actual}$, or at least $P_{online} \subset P_{actual}$. (Of course in an open problem we generally have no way of knowing P_{actual}.) In addition, it is generally assumed that any practical implementation of the algorithms will maintain off-line a set P_{offline} of the best (nondominated) solutions found during the run so far, since stochastic selection methods might cause the loss of nondominated solutions. P_{offline} thus represents the nondominated set of all solutions generated so far during a run. Some algorithms use elitism to ensure that $P_{online} = P_{\text{offline}}$, while others occasionally insert members of P_{offline} back into P_{online} during the run.

3.2.1 Independent Sampling (Multiple Single-Criterion Searches)

One straightforward approach to finding members of P_{actual} is to use multiple single-criterion searches to optimize different aggregations of the criteria. For example, we could try optimizing one criterion at a time. (If successful, this would give us the extrema (corners) of the Pareto-optimal tradeoff surface, e.g., individuals **A** and **G** in Figure 1.). Or we could assume a linear combination (weighted sum) of the objectives, and vary the weights from search to search to gradually build up a sampling of the front.

Fourman (1985), one of the first to perform independent sampling, reports the use of several composite formulae to sample the tradeoff surface. These include linear combinations and lexicographic orderings. In each case Fourman varies the exact formula systematically to obtain different tradeoffs. Ritzel, Eheart, and Ranjithan (1994) discuss multiple GA runs to optimize one criterion at a time, while holding the other criteria constant (using constraints), and then varying the constraint constants to obtain the entire tradeoff surface. Cieniawski (1993) runs a single objective GA several times, using the fitness function $F(\vec{X}) = f_1(\vec{X}) + \alpha f_2(\vec{X})$ (where f_1, f_2 are the two criteria functions). He gradually increases α from zero. Similarly, Tsoi, *et. al.* (1995) and Chang, *et. al.* (1995), both apply a single criterion GA to optimize $F(\vec{x}) = \beta f_1(\vec{x}) + (1-\beta) f_2(\vec{X})$, varying β from zero to one in equal increments, to build up a picture of a two dimensional tradeoff surface of f_1 versus f_2.

Linear aggregative methods, however, are biased toward convex portions of the tradeoff curve. No linear combination exists that will favor points in the concave portions as (global optima). For example, solution **E** in Figure 1 will be inferior to some other member of P no matter what weights are used in the summation. But a *nonlinear* aggregation can be used to sample concave portions. For example, Cieniawski runs multiple single objective GA searches using Tschebycheff's weighting method (discussed above) on a bicriteria problem: $F(\vec{X}) = Max[(1-\beta) * |f_1(\vec{T}) - f_1(\vec{X})|, \beta * |f_2(\vec{T}) - f_2(\vec{X})|]$, varying β from zero to one by 0.05 to obtain both convex and concave portions of the Pareto-optimal frontier.

3.2.2 Cooperative Population Searches

Rather than conduct multiple independent single objective searches, many recent studies have implemented a simultaneous parallel search for multiple members of P_{actual} using a single large population in the hope that the increased implicitly parallel processing (of schemata) will be more efficient and effective. Again, there are several ways to do this, including *criteria selection*, *aggregation selection*, and *Pareto selection.*

3.2.3 Cooperative Population Searches (with Criteria Selection)

Three independent studies (Schaffer, 1984; 1985; Fourman, 1985; Kursawe, 1990; 1991) all implement the same basic idea: parallel single criterion search, or *criteria selection*, in which fractions of the next generation are selected according to one of the k criteria at a time. Probably the first such criteria selection study, and likely also the first multicriteria population-based search in general, was that implemented by Schaffer (1984, 1985), using his *vector evaluated genetic algorithm* (VEGA). VEGA selects a fraction $1/k$ of each new population (next generation) using one of each of the k attributes. (Crossover and mutation are applied to the entire population.) VEGA demonstrated for the first time the successful use of the GA to find multiple members of P using a single population (e.g., Schaffer & Grefenstette, 1985). Fourman (1985), disappointed with the performance of weighted sum and lexicographic ordering aggregations for multiple independent sampling, proposed a selection method similar to VEGA's. Fourman conducts binary tournaments, randomly choosing one criterion to decide each tournament. Later, Kursawe (1990; 1991) implements a randomized criteria selection scheme almost identical to Fourman's. Kursawe suggests that the criteria probabilities be completely random, fixed by the user, or allowed to evolve with the population. He adds a form of *crowding* (De Jong, 1975), as well as dominance and diploidy (Goldberg, 1989), to maintain Pareto diversity. These three similar criteria selection approaches have all been subject to some criticism for being potentially biased against "middling" individuals (i.e., those solutions not excelling at any one particular objective) (Richardson, *et. al.*, 1989; Goldberg, 1989; and empirically, Murata, & Ishibuchi, 1995; Ishibuchi, & Murata, 1996).

3.2.4 Cooperative Population Searches (with Aggregation Selection)

In an attempt to promote more of the "middling" individuals than do the above criteria selection methods, Murata and Ishibuchi (1995; 1996) claim to generalize Kursawe's algorithm. Rather than just use random criteria for selection, their *multi-objective GA* (MOGA) uses random linear combinations of criteria. That is, they randomly vary the weights, $w_i \in [0,1]$ in the summed fitness function $F(\vec{X}) = \sum_{i=0}^{k-1} w_i f_i(\vec{X})$.

3.2.5 Cooperative Population Searches (with Pareto Selection)

Since 1989, several studies have tried to remain truer to the Pareto criterion by using some form of explicit Pareto selection, such that selection favors Pareto optimal solutions (that is, members of P_{online}) above all others, and no preferences are given *within* the Pareto optimal (P_{online}) equivalence class. Many of these efforts have incorporated some form of active *diversity promotion*, such as GA *niching*, to find and maintain an even distribution (sampling) of points along the Pareto front.

Pareto Ranking Goldberg (1989) describes *non-dominated sorting* to rank the population according to Pareto optimality. Under such selection, the currently non-dominated individuals in the population are given rank one and then removed from the population. The newly non-dominated individuals in the reduced population are assigned rank two, and removed. This process continues until all members of the original population are ranked. Goldberg also suggests the use of niching and speciation methods to promote and maintain multiple subpopulations along the Pareto optimal front, but he did not recommend a particular niching method. Goldberg did not implement any of these suggestions at that time.

Hilliard, *et. al* (1989) implement Goldberg's non-dominated sorting, but without niching. Their *Pareto GA* applies proportionate selection to the nondomination ranks. Liepins, *et. al.* (1990) also implement

Goldberg's non-dominated sorting, without niching, but use Baker's (1985) method of rank-based selection (applied to the non-dominated sorting ranks). More recently, Ritzel, *et. al.* (1994) implement Goldberg's non-dominated ranking in their *Pareto GA*, again without niching. They conduct binary tournament selection using the ranks for comparison.

Pareto Ranking Plus Niching (Fitness Sharing) In 1993, four groups independently implemented Goldberg's suggestions for combined Pareto selection and niching (using Goldberg and Richardson's (1987) *fitness sharing*), but in different ways: MOGA, NPGA, NSGA, and the Pareto-optimal Ranking GA with Sharing.

Fonseca and Fleming's (1993a,b; 1994; 1995b,c,d) *Multi-Objective GA* (MOGA) ranks the population according to the "degree of domination": the more members of the current population that dominate a particular individual, the lower its rank. This ranking is more fine-grained than Goldberg's, in that the former can distinguish more ranks than the latter. Apparently, fitness sharing (Goldberg & Richardson, 1987) takes place within each rank only, such that members within each Pareto rank are further ranked according to their fitness sharing *niche counts*. (A niche count is a measure, a direct function, of how crowded the immediate "neighborhood" of an individual is. The more close neighbors, the higher the niche count.) Fonseca and Fleming measure distance (for niche counting) in the *criteria space* (or *attribute space*). Recently, Shaw and Fleming (1996a) apply Fonseca and Fleming's Pareto ranking scheme to a $k = 3$ criteria scheduling problem, both with and without niching (and mating restrictions).

Rather than ranking, Horn and Nafpliotis' (1993, and with Goldberg, 1994) *Niched Pareto GA* (NPGA) implements *Pareto domination tournaments*, binary tournaments using a sample of the current population to determine the dominance status of two competitors **A** and **B**. If one of the competitors is dominated by a member of the sample, and the other competitor is not dominated at all, then the non-dominated individual wins the tournament. If both or neither are dominated, then fitness sharing is used to determine the winner (i.e., whichever has the lower niche count). The sample size (t_{dom}) is used to control "Pareto selection pressure" analogous to the use of tournament size in normal (single objective) tournament selection. The Pareto domination tournament can be seen as a locally calculated, stochastic approximation to Fonseca and Fleming's globally calculated degree-of-domination ranking.

Srinivas and Deb's *Non-dominated Sorting GA* (NSGA) (Srinivas, 1994; Srinivas & Deb, 1995), implements Goldberg's original suggestions as much as possible. NSGA uses Goldberg's suggested Pareto ranking procedure, and incorporates fitness sharing. Unlike MOGA and NPGA, NSGA performs sharing in the phenotypic space (rather than the criteria space), calculating distances between decision variable vectors. Michielssen and Weile (1995) recently combined the nondominated sorting selection of NSGA with the criteria space sharing of MOGA and NPGA.

Eheart, Cieniawski, and Ranjithan (1993) apply Goldberg's nondominated sorting to rank the population, as in NSGA, but then use the ranks as objective fitnesses to be degraded by sharing. (Note that this is different from MOGA, NPGA, and NSGA, which attempt to limit the effects of sharing to competition WITHIN ranks, not between.) Eheart, *et. al.* (1993) then apply tournament selection, using the shared fitnesses, but do not use the "standard" fitness sharing of Goldberg and Richardson (1987). Instead of dividing the objective fitness by the niche count, they add the niche count to the rank. Although they perform sharing in criteria space, as in MOGA and NPGA, they measure distance (i.e., similarity between pairs of individuals) along only one dimension (e.g., "reliability", in their "cost" versus "reliability" bicriteria problem). This biases diversity toward the chosen criterion.

The four efforts above, inspired by Goldberg's 1989 suggestions and incorporating fitness sharing to promote niching within Pareto optimal equivalence classes, are not the only explicitly Pareto selective approaches in the literature. Some of the alternative Pareto selection schemes below implicitly or explicitly maintain at least some diversity in the P-optimal set without using fitness sharing.

Pareto Elitist Recombination Louis and Rawlins (1993) hold four-way Pareto tournaments among two parents and their two (recombined and mutated) offspring. Such a tournament can be seen as the generalization, to multiple objectives, of Thierens and Goldberg's (1994) *elitist recombination* for single objective GAs. The parent-offspring replacement scheme should result in some form of quasi-stable niching (Thierens, 1995). This "parent-offspring non-domination tournament" is applied by Gero and Louis (1995) to beam shape optimization.

Simple Pareto Tournaments with Demes Poloni (1995) holds binary Pareto tournaments, in which an individual that dominates its competitor wins. If neither competitor dominates, a winner is chosen at random. Poloni uses a distributed GA, with multiple small populations, or *demes*, relatively isolated (i.e., little or no migration), as a niching method to try to maintain Pareto diversity. Langdon (1995) generalizes the simple binary Pareto tournament to any number $m \geq 2$ of competitors. A single winner is randomly chosen from the Pareto optimal subset of the m randomly chosen competitors. Langdon favors a steady-state GA, using m-ary Pareto tournaments for deletion selection as well. He also uses demes as a means to maintain diversity. Later (1996), Langdon adds a generalization of Horn and Nafpliotis' (1993) Pareto domination tournaments: if none of the m randomly chosen competitors dominates all of the others, then a separate random sample of the population is used to rank the m competitors. The competitor that is "least dominated" (i.e., dominated by the fewest members of the sample) wins, a stochastic approximation to Fonseca and Fleming's (1993) degree-of-dominance Pareto ranking. Langdon points out correctly that such sampled domination tournaments induce a niching pressure, although he apparently continues to use demes as well (Langdon, 1996).

Pareto Elitist Selection Some researchers recently have implemented *Pareto elitist selection* strategies. These approaches divide the population into just two ranks: dominated and nondominated (i.e., P_{online} and non-P_{online}). Although strongly promoting P_{online}, these algorithms differ in the extent to which they preserve such individuals from one generation to the next. For example, Belegundu, *et. al.* (1994) select only rank one (i.e., P_{online}) for reproduction. (Random individuals are generated to maintain the fixed population size N.) Tamaki, Mori, and Araki (1995) propose a somewhat less severe *Pareto reservation strategy* that copies all non-dominated individuals into the next generation's population. If additional individuals are needed to maintain the fixed population size N, they are selected from the dominated set using criteria selection. Similarly, if $|P_{online}| > N$, then individuals are deleted from the population (i.e., from P_{online}) using "criteria deletion". (Tamaki, *et. al.* (1996) have recently added fitness sharing (in the criteria space) to the Pareto reservation strategy to promote explicitly Pareto diversity (i.e., diversity within P_{online}).)

Other Pareto elitist selection methods include Takada, *et. al.*'s (1996) *Pareto-optimal selection method.* According to Tamaki, *et. al.* (1996), Takada, *et. al.* apply recombination and mutation first, to generate an intermediate population, then select only the Pareto optimal set from among the old and intermediate populations (i.e., all of the parents and offspring). Eheart, *et. al.* (1993) and Cieniawski (1993) use a stochastic approximation to Pareto elitist selection, by maintaining P_{offline} and constantly re-injecting these individuals back into the populations through random replacements (what they call *Pareto-optimal re-injection*). Apparently, they combine this elitism with Pareto optimal rank based tournament selection, with and without niching.

A very unusual Pareto elitist selection method is the *distance method* of Osyczka and Kundu (1995) (applied by Kundu and Kawata (1996), and by Kundu, Kawata, and Watanabe (1996)). The fitness of an individual is a function of its distance to the current Pareto set P_{offline} (as opposed to its distance to an ideal, target vector; see above). This distance is measured in criteria space, using the Euclidean metric, and applying a minimum distance criterion (i.e., the distance from $\vec{X}$ to P_{offline} is equal to the *minimum* of the distances from $\vec{X}$ to any member of P_{offline}). For solutions dominated by P_{offline}, this distance is negative; otherwise, it is positive. The (signed) distance is then added to the individual's fitness. By incorporating this "distance to the nearest member of P_{offline}" into the fitness calculation, the algorithm explicitly promotes criteria-space diversity, both on and off the Pareto frontier, in a manner akin to niching. (Osyczka and Kundu (1996) modify their original method to favor members of P_{online} that dominate members of P_{offline}, to focus search on rapidly improving portions of the tradeoff surface.)

3.3 Integrated Search and Decision Making

A more integrated hybrid of EC search and multicriteria decision making calls for iterative search and decision making. Preliminary multicriteria search is performed to give the DM some idea of the range of tradeoffs possible. The DM then makes some multicriteria decisions to reduce the search space. Additional EC search is limited to this particular region of the criteria space. The iterative process of EC search, multicriteria decisions, EC search, etc. continues until a single solution is left.

Several researchers have suggested such iterative integrations (Horn & Nafpliotis, 1993; Poloni, 1995), but Fonseca and Fleming (1993a) actually implement one: the *goal attainment method*, an extension of their MOGA. In their approach, the original MOGA is run for a few generations, then the DM considers the current P_{offline} and (as above) chooses a target tradeoff point to focus subsequent search. The MOGA is then run for a few more generations using a "modified multiobjective ranking scheme" that considers both Pareto domination and "goal attainment" (e.g., distance-to-target). Fonseca and Fleming briefly discuss the role of the MOGA as a "method for progressive articulation of [DM] preferences".

3.4 State of the Art

Multicriteria EC research has broadened from the early aggregative approaches, with the introduction of criteria selection (e.g., VEGA) in the mid-1980s, the addition of Pareto ranking at the turn of the decade, the combinations of niching and Pareto selection in 1993, and the implementation of many radically different alternative approaches along the way. Two recent reviews (Fonseca & Fleming, 1995a; Tamaki, *et. al.*, 1996) compare some of the latest algorithms with some of the "classics". (In particular, Tamaki, *et. al.* survey several new efforts in Japan which might be inaccessible to non-Japanese readers.) Here we restrict our discussion to a recent trend: hybridization of previously distinct approaches to multicriteria EC optimization.

Most recently (1995, 1996), the new EC conferences include a substantial number of "hybrid" methods, combining old and new techniques for dealing with multiple criteria during EC search. We mention a few example hybrids below.

Hybrid Ordering: Aggregative and Pareto Stanley and Madge (1995) combine Pareto and order aggregative approaches to achieve a very fine-grain ranking. They use a DM's strict ordering of the criteria (assuming one exists) to lexicographically order (and rank) individuals WITHIN each of the ranks produced by Goldberg's nondominated sorting. Fonseca and Fleming (1993a) also use an aggregative method (*goal attainment*, a version of the basic distance-to-target approach) in their MOGA with *progressive articulation of preferences*, to further order the solutions within each of the ranks produced by their degree-of-dominance Pareto ranking scheme discussed earlier. Bhanu and Lee (1994, chapter 9) use the linear combination method (i.e., weighted sum) to reduce a $k = 5$ criteria image segmentation problem to a $k = 2$ bicriteria problem. They sum three quality measures into a single "local quality measure", and the other two quality measures into a "global quality measure". They then apply criteria selection with Pareto elitism. Tsoi, Wong, and Fung (1995) also use weighted sums to combine m pollutant emission levels into a single emissions objective. They optimize the bicriteria problem of cost versus total emissions by applying multiple single-criterion GA runs. Fonseca and Fleming (1994) turn three of five objectives into constraints, and apply their Pareto selective MOGA to find the tradeoff surface of the other two criteria (see Figure 5 of (Fonseca & Fleming, 1994)).

Hybrid Selection: VEGA and Pareto One of Cieniawski's (1993) four multiobjective GA formulations is a *combination VEGA and Pareto-optimal ranking GA*. It is hoped that the combination can find criteria specialists (extremes of the Pareto frontier) using criteria selection, and also favor "middling individuals" in between, using Goldberg's nondominated sorting. To achieve this, Cieniawski simply uses criteria selection for g generations, then switches to the Pareto ranked tournament selection (using Goldberg's nondominated sorting), without niching. He finds that the combination outperforms VEGA and Pareto ranking by themselves, but is similar in performance to Pareto ranking with fitness sharing.

Tamaki, Mori, and Araki (1995) also try to balance VEGA's supposed bias towards Pareto extrema by adding Pareto elitism. Their *non-inferiority preservation strategy* preserves Pareto optimal individuals P_{online} from one generation to the next, unless $|P_{online}| > N$, in which case criteria selection is applied to P_{online} to choose exactly N individuals.

In their COMOGA method (Constrained Optimisation by Multi-Objective Genetic Algorithms), Surrey, Radcliffe, and Boyd (1995) use criteria selection on two criteria: **cost** and **constraints**. When selecting according to a particular criterion, COMOGA implements Fonseca and Fleming's (1993a) Pareto ranking (e.g., **constraints** consists of multiple constraints as "sub-criteria").

Hybrid Search Algorithms As with single criterion EC algorithms, many implementors of multicriteria ECs combine deterministic optimizers (e.g., steepest ascent hillclimbing) or stochastic optimizers (e.g., simulated annealing (SA)) with GA search. Ishibuchi and Murata (1996) add local search to the global selection recombination operators, using their randomly weighted aggregations of attributes (Murata & Ishibuchi, 1995) to hillclimb in multiple directions in criteria space, simultaneously. Tamaki, *et. al.* (1995; 1996) also add local search to their hybrid Pareto reservation strategy with niching and parallel (criteria) selection, applying hillclimbing to each member of the population every 100 generations. Poloni (1995) suggests the use of P_{offline}, found by his Pareto selective GA, by the DM to choose a set of criteria weights and a starting point for subsequent optimization by a domain specific algorithm (the Powell method).

Tsoi, Wong, and Fung (1995) create two GA/SA hybrids (essentially GAs with "cooled" nondeterministic tournament selection) for multiple single-criterion runs to sample P_{actual}. According to Tamaki, *et. al.* (1996), Kita, *et. al.* (1996) apply the SA-like *thermodynamic GA* (TDGA) of Mori, *et. al.* (1995) to a Pareto ranking of the population. The TDGA is designed to maintain a certain level of population "entropy" (i.e., diversity). Combined with Pareto ranking, the TDGA reportedly can maintain Pareto diversity (i.e., large P_{online}).

Hybrid Techniques: (1) Fuzzy Evolutionary Optimization Li, Gen, and Ida (1996) model the uncertainty in the attribute values using fuzzy numbers. They use a GA to optimize a linear combination of the "de-fuzzified" rankings. Their *registered Pareto optimal solution strategy* simply maintains P_{offline}, which is **not** used in selection. Li, Gen, and Ida apply TOPSIS (discussed earlier) to select a "best" member of P_{offline}. **(2) Expert Systems** Fonseca and Fleming (1993a) suggest the use of an automated DM to interact with an interactive multicriteria EC algorithm, using built-in knowledge of a human DM's preferences to guide EC search. An automated DM would make multicriteria decisions on the fly, based on both its DM knowledge base and on accumulating knowledge of the search space and tradeoff surface from the EC algorithm.

3.5 Comparisons: Advantages and Disadvantages

From the range of new and different EC approaches mentioned here, it seems too early to expect rigorous, comprehensive performance comparisons, or even the beginnings of a broad theory of multicriteria EC. More research effort is being spent designing new and hybrid multicriteria EC methods than is going into theory development or controlled experimentation.

3.5.1 Direct, Empirical Comparisons

Most papers on multicriteria EC introduce a new algorithm and compare it with one or two other, well-known approaches (typically VEGA) on a few artificial test problems (e.g., Schaffer's (1985) F2) and one or two open, real-world applications. For example, Hilliard, *et. al.* (1989) compare non-dominated sorting to VEGA and to random search, while Tamaki, *et. al.* (1996) compare VEGA, NPGA, MOGA (Fonseca and Fleming's), and their own Pareto reservation strategy.

3.5.2 Analytical Comparisons

Aside from the scant empirical evidence, we have very little theoretical guidance. We do have conjectures and intuitions, but these sometimes lead to contradictory advice. For example, suppose we want to find P_{actual}. It would seem that a single large cooperative population search for the Pareto front offers greater potential for implicitly parallel, robust search, than do multiple independent searches using aggregations of criteria. On the other hand, multiple independent searches might benefit from the concentration of the population in one particular direction in criteria space. In terms of the goal of the EC search, the Pareto goal might at first seem superior to an aggregative goal, since P_{actual} contains the global optima of all monotonic aggregations of the criteria. But if $|P_{actual}| \gg N$, or if P_{actual} is extremely difficult to find, then the Pareto search might fail, while the aggregative search might find just that one member of P_{actual} that is desired. Then again, even if we can determine prior to search a single aggregation of the criteria that totally orders the search space in complete agreement with the DM's preferences, we might still find that a non-aggregative (e.g., Pareto) search discovers a better optimum than does a single objective search, because of the additional "Pareto diversity" in the population.

We are only beginning to understand how single objective EC algorithms handle nonlinear interactions among decision variables during search. In the multicriteria case, we or our EC algorithms must also deal with nonlinear interactions among criteria. How we choose to handle criteria interactions can change the shape of the entire search space (and vice versa).

3.5.3 Observations

Despite the paucity of empirical comparisons and theory, it seems appropriate to make a few tentative observations based on the current survey of multicriteria EC approaches:

1. Even if the DM agrees strongly with a particular aggregation of criteria prior to a single objective EC search, an additional, Pareto search for the tradeoff surface should be conducted anyway. It might yield a better solution than the single objective search, or it might lead to a re-evaluation of the multicriteria decision made earlier.

2. Most Pareto EC methods, by using P_{online} to check for dominance, induce a selective pressure toward diversity (i.e., an implicit niching effect). But at least for "pure Pareto" approaches this implicit diversity pressure does not exist WITHIN the P_{online}. Without explicit niching, genetic drift takes place within P_{online}. Thus the need for an explicit niching mechanism, such as fitness sharing or crowding, depends on the extent of the implicit diversity pressure. Strong Pareto elitism can lead to a mostly nondominated population and hence genetic drift, for example. There seems to be a need to balance *domination* (selective) pressure with *niching* (diversity) pressure (Horn & Nafpliotis, 1993).

3. Although the Pareto criterion avoids combining or comparing attributes, Pareto EC algorithms, because of their finite populations, MUST make such comparisons. For example, methods that calculate distance in criteria space (e.g., distance to target, distance to P_{offline}, or pairwise distances for niche counts in fitness sharing) combine distances along different attribute dimensions, and thus are highly sensitive to the relative scaling of the attributes (e.g., see (Fonseca & Fleming, 1993; Horn, Nafpliotis, & Goldberg, 1994) for attempts to "normalize" attribute ranges for sharing distance calculations). But any Pareto EC method in general must allocate a finite population along a dense (potentially infinite) Pareto front, thereby choosing particular directions in criteria space on which to focus search. Again, the scaling of the attributes dramatically affects the shape of the current Pareto front as well as the angular difference between search vectors (e.g., exponentially scaling an attribute can change a convex portion of P_{actual} to concave (Cieniawski, 1993)).

4. Criteria selection (e.g., VEGA) and Pareto GAs might be complementary, since the former seem relatively effective at finding extrema of P_{actual} (i.e., solutions that excel at a single criterion), while the latter find many "middling" (compromise) tradeoffs, yet often fail to find or maintain the best "ends" of the Pareto optimal frontier.

5. It is far from clear how well any of the EC approaches scale with the number of criteria. Most applications and proof of principle tests of new algorithms use two objectives (a few exceptions use up to seven). But what of much higher order multicriteria problems? As Fonseca and Fleming (1995a) point out, in general more conflicting objectives mean a flatter partial order, larger P_{actual}, P_{online}, and $P_{\rm offline}$, and less Pareto selective pressure. Non-aggregative EC approaches, and Pareto EC methods in particular, might not scale up well. Hybrid approaches that aggregate SOME of the criteria (above), might help.

6. One major disadvantage of all EC algorithms (or of any stochastic optimizer), compared to enumeration or some other deterministic algorithm, is the same disadvantage non-deterministic search suffers in the single criterion case: we have no way of knowing when to stop searching, since we have no way of knowing if our solution is optimal. In the case of the Pareto approaches this weakness becomes acute. A single solution far off the estimated front can dominate a large portion of the apparent front, drastically changing the "answer" ($P_{\rm offline}$) given by our EC search. Again, this suggests that we not rely on a single multicriteria EC approach, but instead build-up (piece together) a tradeoff surface by taking the best (i.e., Pareto optimal subset) of all solutions discovered by several different methods, EC and non-EC alike (e.g., Ritzel, *et. al.*, 1994).

In general, we are asking our multicriteria EC algorithms to help us decide on our multicriteria preferences as well as to search for the best solutions under those preferences. These two objectives are not separable. Knowing the actual tradeoffs (solutions) available, that is knowing the results of a multicriteria EC search, can help a DM make difficult multicriteria decisions. But making multicriteria decisions up front can help the search for the best solutions. So for now it appears that, as in the early years of single objective EC, practitioners of multicriteria EC are best off experimenting with a number of different methods on a particular problem, both in terms of hybridizing complementary algorithms (above), and in terms of independent checks against each other's performances. On difficult, real-world multicriteria search problems, the most successful approaches will likely by those that (a) incorporate domain and problem specific knowledge, including DM preferences, and (b) use EC searches interactively to build up knowledge of the tradeoffs available and the tradeoffs desired.

4 ALTERNATIVE APPROACHES

It is clear that EC is well-suited to multicriteria problem solving because there are few, if any alternatives. Certainly the field of multiobjective decision analysis offers no competitive search technique, relying mostly on enumeration, or assuming only linear interactions among decision variables, to allow tractable, deterministic search. The traditional alternatives to EC-based optimization (e.g., simulated annealing (SA), stochastic hillclimbing, tabu search, and other robust, stochastic search algorithms) could be substituted for an EC algorithm if the multicriteria problem can be reduced to a single criterion problem, via aggregation. For example, Cieniawski (1993) separately tries simulated annealing, a domain-tailored branch and bound heuristic (MICCP), and a GA, as single objective stochastic optimizers in multiple independent runs using scalar aggregations (both a linear combination and the non-linear Tscbycheff's weighting function) to sample the Pareto front. (He finds that the GA find approximately the same frontier as SA and MICCP.) But if one wishes to perform *multicriteria search*, to determine the entire *Pareto-optimal frontier* (the optimal tradeoff surface), as a number of recent studies have succeeded in doing, then EC algorithms, with their large populations and ability to search *partially ordered* spaces, seem uniquely suited to the task. And they bring to bear the known (if not quantified) benefits of EC search, including implicit and massive parallel processing and tolerance of noisy, uncertain objective functions.

References

Baker, J. E. (1985). Adaptive selection methods for genetic algorithms. In J. J. Grefenstette (Ed.), *Proceedings of an International Conference on Genetic Algorithms and Their Applications (ICGA).* 101–111.

Bhanu, B., & Lee, S. (1994). *Genetic Learning for Adaptive Image Segmentation.* Boston, MA: Kluwer Academic Publishers.

Belegundu, A. D., Murthy, D. V., Salagame, R. R., & Constans, E.W. (1994). Multi-objective optimization of laminated ceramic composites using genetic algorithms. *Proceedings of the 5th AIAA / NASA / USAF / ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Panama City, FL, September 1994. AIAA, Inc. 1015–1022.

Chang, C. S., Wang, W., Liew, A. C., Wen, F. S., & Srinivasan, D. (1995). Genetic algorithm based bicriterion optimisation for traction substations in DC railway system. *1995 IEEE International Conference on Evolutionary Computation (ICEC '95), Volume 1.* Piscataway, NJ: IEEE Service Center. 11–16.

Chipperfield, A. J., Fonseca, C. M., & Fleming, P. J. (1992). Development of genetic optimization tools for multi-objective optimization problems in CACSD. *IEE Colloquium on Genetic Algorithms for Control Systems Engineering, IEE Digest No. 1992/106.* London, UK: The Institution of Electrical Engineers. 3/1–3/6.

Cieniawski, S. E. (1993). *An Investigation of the Ability of Genetic Algorithms to Generate the Tradeoff Curve of a Multi-objective Groundwater Monitoring Problem* (unpublished masters thesis, University of Illinois at Urbana-Champaign, Urbana, IL). [also available as (Eheart, Cieniawski, & Ranjithan, 1993)]

Cieniawski, S. E., Eheart, J. W., & Ranjithan, S. (1995). Using genetic algorithms to solve multiobjective groundwater monitoring problem. *Water Resource Research, 31*(2). 399–409.

de Neufville, R. (1990). *Applied Systems Analysis: Engineering Planning and Technology Management.* McGraw-Hill Publishing Company.

De Jong, K. A. (1975). An analysis of the behavior of a class of genetic adaptive systems. (Doctoral dissertation, University of Michigan). *Dissertation Abstracts International, 36*(10), 5140B. University Microfilms No. 76-9381.

Deb, K. (1989). Genetic algorithms in multimodal function optimization. *Masters thesis, TCGA Report No. 89002.* University of Alabama.

Eheart, J. W., Cieniawski, S. E., & Ranjithan, S. (1993). Genetic algorithm-based design groundwater monitoring system. *WRC Research Report No. 218.* Water Resources Center. University of Illinois at Urbana-Champaign, Urbana, IL. [also available as (Cieniawski, 1993)]

Fonseca, C. M., & Fleming, P. J. (1993a). Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. *Proceedings of the Fifth International Conference on Genetic Algorithms.* Morgan-Kauffman, 416-423.

Fonseca, C. M., & Fleming, P. J. (1993b). Multiobjective genetic algorithms. *IEE.* London, UK: The Institution of Electrical Engineers. 6/1–6/5.

Fonseca, C. M., & Fleming, P. J. (1994). Multiobjective optimal controller design with genetic algorithms. *Proceedings of the IEE Control '94 International Conference, vol. 1 (Warwick, UK)*, IEE Conference Publication No. 389. London, UK: The Institution of Electrical Engineers. 745–749.

Fonseca, C. M., & Fleming, P. J. (1995a). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation, 3*(1). 1–16.

Fonseca, C. M., & Fleming, P. J. (1995b). Multiobjective optimization and multiple constraint handling with evolutionary algorithms I: a unified formulation. *Research Report 564, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 4DU, UK.* January 23, 1995.

Fonseca, C. M., & Fleming, P. J. (1995c). Multiobjective optimization and multiple constraint handling with evolutionary algorithms II: application example. *Research Report 565, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 4DU, UK.* January 23, 1995.

Fonseca, C. M. & Fleming, P. J. (1995d). Multiobjective genetic algorithms made easy: selection, sharing and mating restriction. *Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations of Applications (GALESIA '95*, IEE Conference Publication No. 414. London, UK: The Institution of Electrical Engineers. 44–52.

Fonseca, C. M., & Fleming, P. J. (1996). On the performance assessment and comparison of stochastic multiobjective optimizers. *submitted to Parallel Processing from Nature, 4, Berlin, 1996 (PPSN 4).*

Fourman, M. P. (1985). Compaction of symbolic layout using genetic algorithms. In J. J. Grefenstette (Ed.), *Proceedings of the First International Conference on Genetic Algorithms.* Hillsdale, NJ: Lawrence Erlbaum Associates. 141–153.

Gen, M., Ida, K., & Li, Y. Z. (1995). Solving multiobjective solid transportation problem by genetic algorithms. *Journal of Japan Indus. Manag. Assoc., 46*(5). 446–454. (in Japanese)

Gero, J. S., & Louis, S. J. (1995). Improving Pareto optimal designs using genetic algorithms. *Microcomputers in Civil Engineering, 10.* 239–247.

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

Goldberg, D. E., & Richardson, J. J. (1987). Genetic algorithms with sharing for multimodal function optimization. *Genetic Algorithms and Their Applications: Proceedings of the Second ICGA.* Hillsdale, NJ: Lawrence Erlbaum Associates. 41–49.

Hilliard, M. R., Liepins, G. E., Palmer, M., & Rangarajen, G. (1989). The computer as a partner in algorithmic design: automated discovery of parameters for a multi-objective scheduling heuristic. In R. Sharda, B. L. Golden, E. Wasil, O. Balci, & W. Stewart (Ed.s), *Impacts of Recent Computer Advances on Operations Research.* New York, NY: North-Holland. 321–331.

Horn, J., & Nafpliotis, N. (1993). Multiobjective optimization using the niched Pareto genetic algorithm. *IlliGAL Report No. 93005.* Illinois Genetic Algorithms Laboratory. University of Illinois at Urbana-Champaign.

Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994). A niched Pareto genetic algorithm for multiobjective optimization. *The Proceedings of the First IEEE Conference on Evolutionary Computation (ICEC '94)*, Piscataway, NJ: IEEE Service Center. 82–87.

Hwang, C.-L., Lai, Y.-J., & Liu, T.-Y. (1993). A new approach for multiple objective decision making. *Computers & Operations Research, 20*(8). 889–899.

Ishibuchi, H. & Murata, T. (1996). Multi-objective genetic local search algorithm. In *Proceedings of the Third IEEE International Conference on Evolutionary Computation (ICEC '96).* Piscataway, NJ: IEEE Service Center. 119–124.

Keeney, R., & Raiffa, H., (1976). *Decisions with Multiple Objectives.* New York, NY: Wiley and Sons.

Kita, H., Yabumoto, Y., Mori, N., & Nishikawa, Y. (1996). Multiobjective optimization by means of the thermodynamical genetic algorithm. *submitted to Parallel Processing from Nature, 4, Berlin, 1996 (PPSN 4).*

Kundu, S., & Kawata, S. (1996). AI in control system design using a new paradigm for design representation. In *Proceedings of the Fourth International Conference on Artificial Intelligence in Design.* Boston, MA: Kluwer Academic Publishers. (to appear)

Kundu, S., Kawata, S., & Watanabe, A. (1996). A multicriteria approach to control system design with genetic algorithms. *International Federation of Automatic Control (IFAC) '96 - 13th World Congress, June 30 – July 5, 1996, San Francisco, USA.* Kidlington, UK: Pergamon Press (Elsevier Science imprint). (to appear)

Kursawe, F. (1990). Evolutionsstrategien für die Vektoroptimierung. *Diplomarbeit, Universität Dortmund (Unpublished doctoral dissertation, Dortmund University)*, Dortmund, Germany. (In German)

Kursawe, F. (1991). A variant of evolution strategies for vector optimization. In *Parallel Problem Solving from Nature (PPSN)*, Volume 496 of *Lecture Notes in Computer Science.* Berlin: Springer-Verlag. 193–197.

Langdon, W. B. (1995). Evolving data structures using genetic programming. In L. J. Eshelman (Ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA 6).* San Francisco, CA: Morgan Kaufmann Publishers, Inc. 295–302.

Langdon, W. B. (1996). Using data structures within genetic programming. In J. R. Koza, D. E. Goldberg, D. B. Fogel, & R. L. Riolo (Ed.s), *Genetic Programming 1996.* Cambridge, MA: The MIT Press. 141–149.

Li, Y., Gen, M., & Ida, K. (1996). Evolutionary computation for multicriteria solid transportation problem with fuzzy numbers. In *Proceedings of the Third IEEE International Conference on Evolutionary Computation (ICEC '96).* Piscataway, NJ: IEEE Service Center. 596–601.

Liepins, G. E., Hilliard, M. R., Richardson, J., & Palmer, M. (1990). Genetic algorithms applications to set covering and traveling salesmen problems. In D. E. Brown, & C. C. White (Ed.s), *Operations Research and Artificial Intelligence: The Integration of Problem-Solving Strategies.* Boston, MA: Kluwer Academic Publishers. 29–57.

Louis, S. & Rawlins, G.J. E. (1993). Pareto optimality, GA-easiness, and deception. In S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA 5).* San Mateo, CA: Morgan Kaufmann Publishers. 118–123.

Michielssen, E., & Weile, D. S. (1995). Electromagnetic system design using genetic algorithms. In G. Winter, J. Périaux, M. Galán, & P. Cuesto (Ed.s), *Genetic Algorithms in Engineering and Computer Science.* Chichester, UK: John Wiley & Sons. 345–369 (Chapter 18).

Mori, N., Yoshida, J., Tamaki, H., Kita, H., & Nishikawa, Y. (1995). A thermodynamical selection rule for the genetic algorithm. *1995 IEEE International Conference on Evolutionary Computation (ICEC '95), Volume 1.* Piscataway, NJ: IEEE Service Center. 188–192.

Murata, T., & Ishibuchi, H. (1995). MOGA: multi-objective genetic algorithms. *1995 IEEE International Conference on Evolutionary Computation (ICEC '95), Volume 1.* Piscataway, NJ: IEEE Service Center. 289–294.

Osyczka, A. (1984). *Multicriterion Optimization in Engineering (with FORTRAN programs).* Chichester, UK: Ellis Horwood Limited.

Osyczka, A., & Kundu, S. (1995). A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Structural Optimization 10*(2). 94–99.

Osyczka, A., & Kundu, S. (1996). A modified distance method for multicriteria optimization, using genetic algorithms. *Computers & Industrial Engineering Journal - Special Issue on Genetic Algorithms, 30*(2). (to appear, April 1996)

Pareto, V., (1896). *Cours d'Économie Politique, 1.* Lausanne, Switzerland: F. Rouge.

Poloni, C. (1995). Hybrid GA for multi objective aerodynamic shape optimisation. In G. Winter, J. Périaux, M. Galán, & P. Cuesto (Ed.s), *Genetic Algorithms in Engineering and Computer Science.* Chichester, UK: John Wiley & Sons. 397–415.

Richardson, J. T., Palmer, M. R., Liepins, G., & Hilliard, M. (1989). Some guidelines for genetic algorithms with penalty functions. In J. D. Schaffer (Ed.), *Proceedings of the Third International Conference on Genetic Algorithms.* San Mateo, CA: Morgan-Kaufmann Publishers. 191–197.

Ritzel, B. J., Eheart, W., & Ranjithan, S. (1994). Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. *Water Resources Research, 30*(5). 1589–1603.

Savic, D. A., & Walters, G. A. (1995). Genetic operators and constraint handling for pipe network optimization. In T. C. Fogarty (Ed.), *Evolutionary Computing (AISB Workshop, Sheffield, UK, April 1995), Springer-Verlag Lecture Notes in Computer Science (LNCS) No. 993.* 154–165.

Shaw, K. J., & Fleming, P. J. (1996a). Initial study of multi-objective genetic algorithms for scheduling the production of chilled ready meals. Presented at *The Second International Mendel Conference on Genetic Algorithms (Mendel '96).* (Technical University of Brno, Czech Republic, June 1996). Available as *Research Report No. 623, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 4DU, UK.*

Shaw, K. J., & Fleming, P. J. (1996b). An initial study of practical multi-objective production scheduling, using genetic algorithms. *International Conference on Control '96*, University of Exeter, UK, September 2nd-5th, 1996. (to appear)

Simpson, A. R., Dandy, G. C., & Murphy, L. J. (1994). Genetic algorithms compared to other techniques for pipe optimization. *Journal of Water Resources Planning and Management, 120*(4). 423–443.

Srinivas, N. (1994). *Multiobjective optimization using nondominated sorting in genetic algorithms* (unpublished masters thesis, Indian Institute of Technology, Kanpur, India).

Srinivas, N., & Deb, K. (1995). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation, 2*(3). 221–248.

Stanley, T. J., & Mudge, T. (1995). A parallel genetic algorithm for multiobjective microprocessor design. In L. J. Eshelman (Ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA 6).* San Francisco, CA: Morgan Kaufmann Publishers, Inc. 597–604.

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

Surrey, P. D., Radcliffe, N. J., & Boyd, I. D. (1995). A multi-objective approach to constrained optimisation of gas supply networks: the COMOGA method. In T. C. Fogarty (Ed.), *Evolutionary Computing (AISB Workshop, Sheffield, UK, April 1995), Springer-Verlag Lecture Notes in Computer Science (LNCS) No. 993.* Berlin: Springer-Verlag. 166–180.

Takada, Y., Yamamura, M., & Kobayashi, S. (1996). An approach to portfolio selection problems using multi-objective genetic algorithms. *Proceedings of the 23rd Symposium on Intelligent Systems.* 103–108. (in Japanese)

Tamaki, H., Kita, H. & Kobayashi, S. (1996). Multi-objective optimization by genetic algorithms: a review. In *Proceedings of the Third IEEE International Conference on Evolutionary Computation (ICEC '96).* Piscataway, NJ: IEEE Service Center. 517–522.

Tamaki, H., Mori, M., & Araki, M. (1995). Generation of a set of Pareto-optimal solutions by genetic algorithms. *Transactions of the Society of Instrument and Control Engineers, 31*(8). 1185–1192. (in Japanese)

Tamaki, H., Mori, M., Araki, M., Mishima, Y., & Ogai, H. (1995). Multicriteria optimization by genetic algorithms: a case of scheduling in hot rolling process. In M. Fushimi, & K. Tone (Ed.s), *Proceedings of the Third Conference of the Association of Asian-Pacific Operational Research Societies (APORS '94).* Singapore: World Scientific. 374–381.

Thierens, D., & Goldberg, D. E. (1994). Elitist recombination: an integrated selection recombination GA. *The Proceedings of the First IEEE Conference on Evolutionary Computation (ICEC '94)*, Piscataway, NJ: IEEE Service Center. 508–512.

Thierens, D. (1995). Analysis and design of genetic algorithms, *Unpublished doctoral dissertation, Catholic University of Leuven, Leuven, Belgium.*

Tsoi, E., Wong, K. P., & Fung, C. C. (1995). Hybrid GA/SA algorithms for evaluating trade-off between economic cost and environmental impact in generation dispatch. *1995 IEEE International Conference on Evolutionary Computation (ICEC '95), Volume 1.* Piscataway, NJ: IEEE Service Center. 132–137.

Vemuri, V. R., & Cedeño, W. (1995). A new genetic algorithm for multi-objective optimization in water resource management. *1995 IEEE International Conference on Evolutionary Computation (ICEC '95), Volume 1.* Piscataway, NJ: IEEE Service Center. 495–500.

Wallace, D. R., Jakiela, M. J., & Flowers, W. C. (1994). Design search under probabilistic specifications using genetic algorithms. *Computer-Aided Design, 28*(5). 405–420.

Wienke, D., Lucasius, C., & Kateman, G. (1992). Multicriteria target vector optimization of analytical procedures using a genetic algorithm (Part I. Theory, numerical simulations and application to atomic emission spectroscopy). *Analytica Chimica Acta, 265*(2). 211–225.

Yoshida, K., Yamamura, M., & Kobayashi, S. (1996). Generating a set of Pareto optimal decision trees by GA. *Proceedings of the 4th International Conference on Soft Computing, 1996.* (to appear)