

## F1.9 Multicriterion decision making

*Jeffrey Horn*

### Abstract

Applying evolutionary computation (EC) to multicriterion decision making addresses two difficult problems: (i) searching intractably large and complex spaces and (ii) 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 multicriterion 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 multicriterion decisions allows for two straightforward hybrid strategies: (i) make multicriterion decisions *first*, to aggregate objectives, then apply EC search to optimize the resulting figure of merit, or (ii) conduct multiple EC searches *first* using different aggregations of the objectives in order to obtain a range of alternative solutions, then make a multicriterion 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 nondominated solutions. A few researchers have further suggested ways of *integrating* multicriterion decision making and EC search, by iteratively using EC search to sample the tradeoff surface while using multicriterion decision making to successively narrow the search. Although all these approaches have received only limited testing and analysis, there are few comparable alternatives to multicriterion EC search (for searching intractably large spaces to meet multiple criteria).

### F1.9.1 Introduction

One could argue that real-world problems are, in general, multicriterion. That is, the problems involve multiple objectives to be met or optimized, with the objectives often conflicting. (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).) The application of evolutionary search to multicriterion problems seems a logical next step for the evolutionary computation (EC) approaches that have been successful on hard single-criterion problems. Indeed, quite a few EC approaches have found very satisfactory ‘tradeoff solutions’ in multiobjective problems. However EC search can be, and has been, applied to multiobjective problems in a number of very different ways. It is far from clear which, if any, approaches are superior for general classes of multiobjective problems. At this early point in the development of multicriterion ECs, it would be a good idea to try more than one approach on any given problem.

## F1.9.2 Description of the multicriterion application domain

Multicriterion problems in general involve two ‘quasiseparable’ types of problem difficulty: *search* and *multicriterion 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 monotonic aggregation of the multiple objectives into a single objective will maintain this ordering.)

Traditionally these two aspects of the overall problem, search and multicriterion 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 multicriterion decision, not search, is the focus of analysis. EC is in a unique position to address *both* search and multicriterion 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 *multicriterion search problem*, suited for multicriterion EC optimization.

D3.5.2, D3.5.4

### F1.9.2.1 Practical applications

Multicriterion 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 multicriterion 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, Krause and Nissen 1995). 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 multicriterion problem, with each DM’s ranking (rating or ordering) being treated as a separate criterion.

### F1.9.2.2 The search space

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

Search and multicriterion decisions are not independent tasks. Making some multicriterion choices before search can alter the ‘fitness landscape’ of the search space by adding more ‘ordering information’, 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 multicriterion 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.

## F1.9.3 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 multicriterion 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 multicriterion decisions. At the highest level, there are three general orderings for conducting search and making multicriterion decisions:

- (i) make multicriterion decisions *before* search (decide  $\Rightarrow$  search),
- (ii) search *before* making multicriterion decisions (search  $\Rightarrow$  decide), and
- (iii) integrate search and multicriterion decision making (decide  $\Leftrightarrow$  search).

### F1.9.3.1 Multicriterion 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 (nonscalar)* approaches.

*The scalar-aggregative approach.* The most common aggregative methods combine the various objectives into a single scalar-valued *utility function*,  $U(\mathbf{A})$ , where  $U : \mathbb{R}^k \rightarrow \mathbb{R}$ , reflecting the multicriterion 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 method, such as fitness-proportionate selection (e.g. roulette wheel, stochastic remainder), although other selection methods require only a complete ordering (e.g. linear ranking) or merely a partial ordering (e.g. tournament selection).

The simplest example of a scalar aggregation is a *linear combination* (i.e. *weighted sum*), such as  $U(\mathbf{A}) = w_0a_0 + w_1a_1 + \dots + w_{k-1}a_{k-1}$ , 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, chs 4, 8) sum five measures of image segmentation quality into a single objective using equal weights, while Vemuri and Cedeño (1995) first rank the population  $k$  times using each of the criteria, then for each solution sum the  $k$  criterion 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 (see e.g. Simpson *et al* 1994, Savic and Walters 1995, Krause and Nissen 1995). Richardson *et al* (1989) give guidelines for using penalty functions with GAs, while Stanley and Mudge (1995) discuss turning constraints ‘back into’ objectives. C5  
C5.2

A recent development in the decision analysis community handles multiplicative nonlinearities: *multiattribute utility analysis (MAUA)* (Keeney and 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 pairwise 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 *et al* (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  $\mathbf{T}$  is chosen as an ideal solution to shoot for. Solutions are evaluated by simply measuring their distance from this theoretical goal in criterion space (see e.g. Wienke *et al* 1992). Choosing the target vector and the form of the metric both involve multicriterion 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,

$$h_p(\mathbf{A}, \mathbf{B}) = \left( \sum_{i=0}^{k-1} |a_i - b_i|^p \right)^{1/p} \quad p \geq 1. \quad (\text{F1.9.1})$$

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  $k$ -criterion problem to a  $k = 2$ , bicriterion one.) Hwang *et al* (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, and Deb 1995). Minimax seeks to minimize the maximum ‘criterion distance’ to the target solution  $\mathbf{T}$ . Choosing the maximum of the  $k$  criterion distances is equivalent to using the *maximum Holder metric*, which is obtained as  $p \rightarrow \infty$  in equation (F1.9.1) above:  $h_\infty(\mathbf{A}, \mathbf{T}) = \max(|a_0 - t_0|, |a_1 - t_1|, \dots, |a_{k-1} - t_{k-1}|)$ . Minimizing this distance becomes the single objective. By differentially scaling the individual criterion 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 ‘Independent sampling’ in section F1.9.3.2 below).

*The order-aggregative approach.* 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. Nonscalar, order-aggregative methods (e.g. lexicographic ordering or voting schemes) impose a total ordering on the space of solutions, but do not provide any meaningful scalar evaluation useful for a fitness function. Therefore such methods are best suited for rank-based selection, including tournament selection. C2.3

*F1.9.3.2 Search before multicriterion 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 noncommensurate? Recognizing the difficulty, perhaps the impossibility, of making all of the multicriterion decisions up front, many users and researchers have chosen to first apply search to find a set of ‘best alternatives’. Multicriterion decision-making methods can then be applied to the reduced set of solutions.

Vilfredo Pareto (1896) recognized that, even without making any multicriterion 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  $(a_0, a_1, a_2, \dots, a_{k-1})$ , and a solution B, with attribute values  $(b_0, b_1, b_2, \dots, b_{k-1})$ , 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 multicriterion 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 multicriterion decisions. This *nondominated set* is known by many names: the *Pareto-optimal set*, the *admissible set*, the *efficient points*, the *nondominated frontier*, and the *Pareto front*, for example. B2.1.3

The terms *front* and *frontier* arise from the geometric depiction of the *criterion space* (or *attribute space*). Figure F1.9.1 depicts a  $k = 2$ -criterion space. Here the chosen criteria are cost (to be minimized)

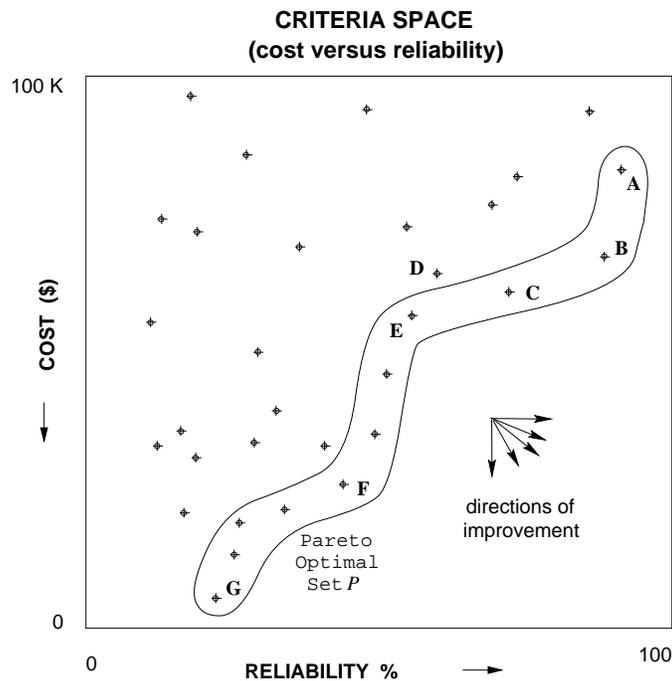


Figure F1.9.1. Criterion space in an example two-criterion problem.

and reliability (to be maximized). A population of individuals (candidate solutions) is plotted using each individual's evaluated criterion vector as a coordinate. Note that in figure F1.9.1 individual C dominates D, but does not dominate E (they are incomparable). 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 multicriterion decision-making process for several reasons. (i) Knowledge of the nature of  $P$  might simplify the multicriterion 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 F1.9.1), or perhaps a 'knee' of the front (e.g. B or F in figure F1.9.1), at which large sacrifices in one attribute yield only small improvements in the other(s). (ii)  $P$  is 'DM independent'. If a DM (or his or her preferences) changes, the Pareto search need not be performed again. (iii) 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 and Fleming 1993a, b, Horn and Nafpliotis 1993). Thus, Pareto approaches allow the study of 'tradeoffs', not just solutions.

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; we call this  $P_{\text{actual}}$ . Any given population (e.g. generation) has a nondominated subset of individuals,  $P_{\text{on-line}}$ . The hope is that by the end of the run,  $P_{\text{on-line}} = P_{\text{actual}}$ , or at least  $P_{\text{on-line}} \subset P_{\text{actual}}$ . (Of course in an open problem we generally have no way of knowing  $P_{\text{actual}}$ .) In addition, it is generally assumed that any practical implementations of the algorithms will maintain off-line a set  $P_{\text{off-line}}$  of the best (nondominated) solutions found during the run so far, since stochastic selection methods might cause the loss of nondominated solutions.  $P_{\text{off-line}}$  thus represents the nondominated set of all solutions generated so far during a run. Some algorithms use elitism to ensure that  $P_{\text{on-line}} = P_{\text{off-line}}$ , while others occasionally insert members of  $P_{\text{off-line}}$  back into  $P_{\text{on-line}}$  during the run.

*Independent sampling (multiple single-criterion searches).* One straightforward approach to finding members of  $P_{\text{actual}}$  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 F1.9.1.)

Alternatively 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 *et al* (1994) discuss multiple GA runs to optimize one criterion at a time, while holding the other criteria constant (using constraints). They then vary the constraint constants to obtain the entire tradeoff surface.

Cieniawski (1993) runs a single-objective GA several times, using the fitness function  $F(\mathbf{X}) = f_1(\mathbf{X}) + \alpha f_2(\mathbf{X})$  (where  $f_1$  and  $f_2$  are the two criterion functions). He gradually increases  $\alpha$  from zero. Similarly, Tsoi *et al* (1995) and Chang *et al* (1995) both apply a single-criterion GA to optimize  $F(\mathbf{x}) = \beta f_1(\mathbf{x}) + (1 - \beta) f_2(\mathbf{x})$ , varying  $\beta$  from zero to one in equal increments, to build up a picture of a two-dimensional tradeoff surface of  $f_1$  versus  $f_2$ . Note that the number of sample points needed to maintain a constant sampling density increases exponentially in  $k$ .

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 F1.9.1 will be inferior to some other member of  $P$  no matter what weights are used in the summation. However 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 bicriterion problem:  $F(\mathbf{X}) = \max[(1 - \beta)|f_1(\mathbf{T}) - f_1(\mathbf{X})|, \beta|f_2(\mathbf{T}) - f_2(\mathbf{X})|]$ , varying  $\beta$  from zero to one by 0.05 to obtain both convex and concave portions of the Pareto-optimal frontier.

*Cooperative population searches.* Rather than conduct multiple independent single-objective searches, many recent studies have implemented a simultaneous parallel search for multiple members of  $P_{\text{actual}}$  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 *criterion selection*, *aggregation selection*, and *Pareto selection*.

*Cooperative population searches (with criterion selection).* Three independent studies (Schaffer 1984, 1985, Fourman 1985, Kursawe 1990, 1991) all implement the same basic idea: parallel single-criterion search, or *criterion selection*, in which fractions of the next generation are selected according to one of the  $k$  criteria at a time. Probably the first such criterion selection study, and probably also the first multicriterion 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 (see e.g. Schaffer and 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) implemented a randomized criterion selection scheme almost identical to Fourman's. Kursawe suggests that the criterion 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 criterion 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 and Ishibuchi 1995, Krause and Nissen 1995, Ishibuchi and Murata 1996).

C6.1.3

*Cooperative population searches (with aggregation selection).* In an attempt to promote more of the 'middling' individuals than do the above criterion selection methods, Murata and Ishibuchi (1995, Ishibuchi and Murata 1996) claim to generalize Kursawe's algorithm. Rather than just use random criteria for selection, their *multiobjective 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  $F(\mathbf{X}) = \sum_{i=0}^{k-1} w_i f_i(\mathbf{X})$ .

*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  $P_{\text{on-line}}$ ) above all others, and no preferences are given *within* the Pareto-optimal ( $P_{\text{on-line}}$ ) 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. C6.1

*Pareto ranking.* Goldberg (1989) describes *nondominated sorting* to rank the population according to Pareto optimality. Under such selection, the currently nondominated individuals in the population are given rank one and then removed from the population. The newly nondominated 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 suggested 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 nondominated sorting, but without niching. Their *Pareto GA* applies proportionate selection to the nondomination ranks. Liepins, *et al* (1990) also implement Goldberg's nondominated sorting, without niching, but use Baker's (1985) method of rank-based selection (applied to the nondominated sorting ranks). More recently, Ritzel *et al* (1994) have implemented Goldberg's nondominated sorting 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 the *fitness sharing* of Goldberg and Richardson (1987)), but in different ways: MOGA, NPGA, NSGA, and the Pareto-optimal ranking GA with sharing. C6.1.2

The *multiobjective GA* (MOGA) of Fonseca and Fleming (1993a, b, 1994, 1995b–d) 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 finer grained than Goldberg's, in that the former can distinguish more ranks than the latter. (Note that any method of ranking a partially ordered set allows the use of traditional rank-based selection schemes on Pareto-ordered spaces.) Apparently, fitness sharing (Goldberg and 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 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 *criterion space* (or *attribute space*). Recently, Shaw and Fleming (1996a) have applied the Pareto ranking scheme of Fonseca and Fleming to a  $k = 3$ -criterion scheduling problem, both with and without niching (and mating restrictions).

Rather than ranking, the *niched Pareto GA* (NPGA) of Horn and Nafpliotis (1993, Horn *et al* 1994) 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 nondominated 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 ( $t_{\text{dom}}$ ) is used to control 'Pareto selection pressure' analogously 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 the globally calculated degree-of-domination ranking of Fonseca and Fleming.

The *non-dominated sorting GA* (NSGA) of Srinivas and Deb (Srinivas 1994, Srinivas and 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 criterion space), calculating distances between decision variable vectors. Michielssen and Weile (1995) recently combined the nondominated sorting selection of NSGA with the criterion space sharing of MOGA and NPGA.

Eheart *et al* (1993) and Cieniawski *et al* (1985) 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.) They 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 criterion 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’ bicriterion 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 the *elitist recombination* of Thierens and Goldberg (1994) for single-objective GAs. The parent–offspring replacement scheme should result in some form of quasistable niching (Thierens 1995). This ‘parent–offspring nondomination tournament’ is applied by Gero and Louis (1995) to beam shape optimization, and generalized to  $\mu$  parents and  $\lambda$  offspring in a  $(\mu + \lambda)$ -ES by Krause and Nissen (1995).

*Simple Pareto tournaments with demes.* Poloni (1995) and Baita *et al* (1995) hold 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, Langdon (1996) adds a generalization of the Pareto domination tournaments of Horn and Nafpliotis (1993): 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 the degree-of-dominance Pareto ranking of Fonseca and Fleming (1993a). 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.  $P_{\text{on-line}}$  and non- $P_{\text{on-line}}$ ). Although strongly promoting  $P_{\text{on-line}}$ , 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.  $P_{\text{on-line}}$ ) for reproduction. (Random individuals are generated to maintain the fixed population size  $N$ .) Tamaki *et al* (1995) propose a somewhat less severe *Pareto reservation strategy* that copies all nondominated 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 criterion selection. Similarly, if  $|P_{\text{on-line}}| > N$ , then individuals are deleted from the population (i.e. from  $P_{\text{on-line}}$ ) using ‘criterion deletion’. (Tamaki *et al* (1996) have recently added fitness sharing (in the criterion space) to the Pareto reservation strategy to promote explicitly Pareto diversity (i.e. diversity within  $P_{\text{on-line}}$ .) Applications of their approach can be found in Tamaki *et al* (1995) and Yoshida *et al* (1996).

Other Pareto elitist selection methods include the *Pareto-optimal selection method* of Takada *et al* (1996). 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). Krause and Nissen (1995) implement the same ‘ $(\mu + \lambda)$  Pareto elitism selection’ as Takada *et al* but use an ES rather than a GA. Eheart *et al* (1993) and Cieniawski (1993) use a stochastic approximation to Pareto elitist selection, by maintaining  $P_{\text{off-line}}$  and constantly reinjecting these individuals back into the populations through random replacements (what they call *Pareto-optimal reinjection*). 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 *et al* (1996)). The fitness of an individual is a function of its distance from the current Pareto set  $P_{\text{off-line}}$  (as opposed to its distance from an ideal, target vector; see above). This distance is measured in criterion space, using the Euclidean metric, and applying a minimum-distance criterion (i.e. the distance from  $X$  to  $P_{\text{off-line}}$  is equal to the *minimum* of the distances from  $X$  to any member of  $P_{\text{off-line}}$ ). For solutions dominated by  $P_{\text{off-line}}$ , 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  $P_{\text{off-line}}$ ' into the fitness calculation, the algorithm explicitly promotes criterion-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  $P_{\text{on-line}}$  that dominate members of  $P_{\text{off-line}}$ , to focus search on rapidly improving portions of the tradeoff surface.)

### F1.9.3.3 Integrated search and decision making

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

Several researchers have suggested such iterative integrations (Horn and 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  $P_{\text{off-line}}$  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'.

### F1.9.3.4 State of the art

Multicriterion EC research has broadened from the early aggregative approaches, with the introduction of criterion 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 and 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 multicriterion EC optimization.

Most recently (1995, 1996), the 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 approaches.* Stanley and Mudge (1995) combine Pareto and order aggregative approaches to achieve a very fine-grained 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. More recently, Greenwood *et al* (1997) relax the need for a strict ordering of the criteria to permit the DM to perform only an 'imprecise ranking of attributes'. 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, ch 9) use the linear combination method (i.e. weighted sum) to reduce a  $k = 5$ -criterion image segmentation problem to a  $k = 2$  bicriterion 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 criterion selection with Pareto elitism. Tsoi *et al* (1995) also use weighted sums to combine  $m$  pollutant emission levels into a single emissions objective. They optimize the bicriterion 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 and Fleming 1994).

*Hybrid selection: VEGA and Pareto approaches.* 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 criterion specialists (extremes of the Pareto frontier) using criterion selection, and also favor 'middling individuals' in between, using Goldberg's nondominated sorting. To achieve this, Cieniawski simply uses criterion 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 *et al* (1995) also try to balance VEGA's supposed bias towards Pareto extrema by adding Pareto elitism. Their *noninferiority preservation strategy* preserves Pareto-optimal individuals  $P_{\text{on-line}}$  from one generation to the next, unless  $|P_{\text{on-line}}| > N$ , in which case criterion selection is applied to  $P_{\text{on-line}}$  to choose exactly  $N$  individuals.

In their COMOGA method (constrained optimization by multiobjective genetic algorithms), Surrey *et al* (1995) use criterion selection on two criteria: *cost* and *constraints*. When selecting according to a particular criterion, COMOGA implements the Pareto ranking of Fonseca and Fleming (1993a) (e.g. *constraints* consists of multiple constraints as 'subcriteria').

*Hybrid search algorithms.* As with single-criterion EC algorithms, many implementors of multicriterion 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 and Ishibuchi 1995) to hillclimb in multiple directions in criterion space, simultaneously. Tamaki *et al* (1995, 1996) also add local search to their hybrid Pareto reservation strategy with niching and parallel (criterion) selection, applying hillclimbing to each member of the population every 100 generations. Poloni (1995) suggests the use of  $P_{\text{off-line}}$ , found by his Pareto selective GA, by the DM to choose a set of criterion weights and a starting point for subsequent optimization by a domain-specific algorithm (the Powell method).

Tsoi *et al* (1995) create two GA-SA hybrids (essentially GAs with 'cooled' nondeterministic tournament selection) for multiple single-criterion runs to sample  $P_{\text{actual}}$ . 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  $P_{\text{on-line}}$ ).

#### Hybrid techniques

- (i) *Fuzzy evolutionary optimization.* Li *et al* (1996) model the uncertainty in the attribute values using fuzzy numbers. They use a GA to optimize a linear combination of the 'defuzzified' rankings. Their *registered Pareto-optimal solution strategy* simply maintains  $P_{\text{off-line}}$ , which is *not* used in selection. Li *et al* apply TOPSIS (discussed earlier) to select a 'best' member of  $P_{\text{off-line}}$ . D2
- (ii) *Expert Systems.* Fonseca and Fleming (1993a) suggest the use of an automated DM to interact with an interactive multicriterion EC algorithm, using built-in knowledge of a human DM's preferences to guide EC search. An automated DM would make multicriterion 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.

#### F1.9.4 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 multicriterion EC. More research effort is being spent designing new and hybrid multicriterion EC methods than is going into theory development or controlled experimentation.

*Direct, empirical comparisons.* Most papers on multicriterion 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 nondominated sorting to VEGA and to random search, while Tamaki *et al* (1996) compare VEGA, NPGA, MOGA (Fonseca and Fleming), and their own Pareto reservation strategy.

*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  $P_{\text{actual}}$ . 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 criterion space. In terms of the goal of the EC search, the Pareto goal might at first seem superior to an aggregative goal, since  $P_{\text{actual}}$  contains the global optima of all monotonic aggregations of the criteria. However if  $|P_{\text{actual}}| \gg N$ , or if  $P_{\text{actual}}$  is extremely difficult to find, then the Pareto search might fail, while the aggregative search might find just that one member of  $P_{\text{actual}}$  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 nonaggregative (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 multicriterion case, we or our EC algorithms must also deal with nonlinear interactions among criteria. How we choose to handle criterion interactions can change the shape of the entire search space (and vice versa).

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

- (i) 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 reevaluation of the multicriterion decision made earlier.
- (ii) Most Pareto EC methods, by using  $P_{\text{on-line}}$  to check for dominance, induce a selective pressure toward diversity (i.e. an implicit niching effect), but at least for 'pure Pareto' approaches (e.g. nondominated sorting) this implicit diversity pressure does not exist *within* the  $P_{\text{on-line}}$ . Without explicit niching, genetic drift takes place within  $P_{\text{on-line}}$ . 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 and Nafpliotis 1993).
- (iii) 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 criterion space (e.g. distance to target, distance to  $P_{\text{off-line}}$ , 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 (see e.g. Fonseca and Fleming 1993a and Horn *et al* 1994) for attempts to 'normalize' attribute ranges for sharing distance calculations). However any Pareto EC method in general must allocate a finite population along a dense (potentially infinite) Pareto front, thereby choosing particular directions in criterion 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 portion of  $P_{\text{actual}}$  from convex to concave (Cieniawski 1993)).
- (iv) Criterion selection (e.g. VEGA) and Pareto GAs might be complementary, since the former seem relatively effective at finding extrema of  $P_{\text{actual}}$  (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 (e.g. Krause and Nissen 1995).
- (v) 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 multicriterion problems? As Fonseca and Fleming (1995a) point out, in general more conflicting objectives mean a flatter partial order, larger  $P_{\text{actual}}$ ,  $P_{\text{on-line}}$ , and  $P_{\text{off-line}}$ , and less Pareto selective pressure. Nonaggregative EC approaches, and Pareto EC methods in particular, might not scale up well. Hybrid approaches that aggregate *some* of the criteria (above) might help.

- (vi) One major disadvantage of all EC algorithms (or of any stochastic optimizer), compared to enumeration or some other deterministic algorithm, is the same disadvantage nondeterministic search suffers in the single-criterion case: we have no way of knowing when to stop searching, since we have no way of knowing whether 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’ ( $P_{\text{off-line}}$ ) given by our EC search. Again, this suggests that we not rely on a single multicriterion 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 (see e.g. Ritzel *et al* 1994).

In general, we are asking our multicriterion EC algorithms to help us decide on our multicriterion 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 multicriterion EC search, can help a DM make difficult multicriterion decisions, but making multicriterion decisions up front can help the search for the best solutions. For now it appears that, as in the early years of single-objective EC, practitioners of multicriterion 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 multicriterion search problems, the most successful approaches will probably be those that (i) incorporate domain- and problem-specific knowledge, including DM preferences, and (ii) use EC searches interactively to build up knowledge of the tradeoffs available and the tradeoffs desired.

### F1.9.5 Alternative approaches

It is clear that EC is well suited to multicriterion 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 multicriterion problem can be reduced to a single-criterion problem, via aggregation. For example, Cieniawski (1993) separately tries SA, 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 nonlinear Tschycheff weighting function) to sample the Pareto front. (He finds that the GA finds approximately the same frontier as SA and MICCP.) However if one wishes to perform *multicriterion 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. G9.7.4

### Acknowledgement

Most of the research for this contribution was performed by the author while in residence at the Illinois Genetic Algorithms Laboratory (IlligAL) at the University of Illinois at Urbana-Champaign (UIUC). The author acknowledges support provided by NASA under contract number NGT-50873, the US Army under contract DASG60-90-C-0153, and the National Science Foundation under grant ECS-9022007, while at the IlligAL at UIUC.

### References

- Baita F, Mason F, Poloni C and Ukovich W 1995 Genetic algorithm with redundancies for the vehicle scheduling problem *Evolutionary Algorithms for Management Applications* ed J Biethahn and V Nissen (Berlin: Springer) pp 341–53

- Baker J E 1985 Adaptive selection methods for genetic algorithms *Proc. Int. Conf. on Genetic Algorithms and Their Applications (ICGA)* ed J J Grefenstette (Hillsdale, NJ: Erlbaum) pp 101–11
- Bhanu B and Lee S 1994 *Genetic Learning for Adaptive Image Segmentation* (Boston, MA: Kluwer)
- Belegundu A D, Murthy D V, Salagame R R and Constans E W 1994 Multi-objective optimization of laminated ceramic composites using genetic algorithms *Proc. 5th AIAA/NASA/USAF/ISSMO Symp. on Multidisciplinary Analysis and Optimization (Panama City, FL, 1994)* (AIAA) pp 1015–22
- Chang C S, Wang W, Liew A C, Wen F S and Srinivasan D 1995 Genetic algorithm based bicriterion optimisation for traction substations in DC railway system *Proc. 2nd IEEE Int. Conf. on Evolutionary Computation (Perth, November–December 1995)* vol 1 (Piscataway, NJ: IEEE) pp 11–16
- Cieniawski S E 1993 *An Investigation of the Ability of Genetic Algorithms to Generate the Tradeoff Curve of a Multi-objective Groundwater Monitoring Problem* Master's Thesis, University of Illinois at Urbana-Champaign
- Cieniawski S E, Eheart J W and Ranjithan S 1995 Using genetic algorithms to solve multiobjective groundwater monitoring problem *Water Resource Res.* **31** 399–409
- De Jong K A 1975 *An Analysis of the Behavior of a Class of Genetic Adaptive Systems* Doctoral Dissertation, University of Michigan; *Dissertation Abstracts Int.* **36** 5140B; University Microfilms 76-9381
- de Neufville R 1990 *Applied Systems Analysis: Engineering Planning and Technology Management* (New York: McGraw-Hill)
- Eheart J W, Cieniawski S E and Ranjithan S 1993 *Genetic Algorithm-Based Design Groundwater Monitoring System* WRC Research Report 218, Water Resources Center, University of Illinois at Urbana-Champaign
- Fonseca C M and Fleming P J 1993a Genetic algorithms for multiobjective optimization: formulation, discussion and generalization *Proc. 5th Int. Conf. on Genetic Algorithms (Urbana-Champaign, IL, July 1993)* ed S Forrest (San Mateo, CA: Morgan Kaufmann) pp 416–23
- 1993b *Multiobjective Genetic Algorithms* (London: IEE) pp 6/1–6/5
- 1994 Multiobjective optimal controller design with genetic algorithms *Proc. IEE Control '94 Int. Conf. (Warwick, 1994)* vol 1 (*IEE Conf. Publication 389*) (London: IEE) pp 745–9
- 1995a An overview of evolutionary algorithms in multiobjective optimization *Evolutionary Comput.* **3** 1–16
- 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
- 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
- 1995d Multiobjective genetic algorithms made easy: selection, sharing and mating restriction *Proc. 1st IEE/IEEE Int. Conf. on Genetic Algorithms in Engineering Systems: Innovations of Applications (GALESIA '95)* (*IEE Conf. Publication 414*) (London: IEE) pp 44–52
- Fourman M P 1985 Compaction of symbolic layout using genetic algorithms *Proc. 1st Int. Conf. on Genetic Algorithms (Pittsburgh, PA, July 1985)* ed J J Grefenstette (Hillsdale, NJ: Erlbaum) pp 141–53
- Gero J S and Louis S J 1995 Improving Pareto optimal designs using genetic algorithms *Microcomput. Civil Eng.* **10** 239–47
- Goldberg D E 1989 *Genetic Algorithms in Search, Optimization, and Machine Learning* (Reading, MA: Addison-Wesley)
- Goldberg D E and Richardson J J 1987 Genetic algorithms with sharing for multimodal function optimization *Proc. 2nd Int. Conf. on Genetic Algorithms (Cambridge, MA, 1987)* ed J J Grefenstette (Hillsdale, NJ: Erlbaum) pp 41–9
- Greenwood G, Hu X and D'Ambrosio J G 1997 Fitness functions for multiobjective optimization problems: combining preferences with Pareto rankings *Foundations of Genetic Algorithms, IV* ed R K Belew and M D Vose (San Francisco, CA: Morgan Kaufmann) at press
- Hilliard M R, Liepins G E, Palmer M and Rangarajen G 1989 The computer as a partner in algorithmic design: automated discovery of parameters for a multi-objective scheduling heuristic *Impacts of Recent Computer Advances on Operations Research* ed R Sharda, B L Golden, E Wasil, O Balci and W Stewart (New York: North-Holland) pp 321–31
- Horn J and Nafpliotis N 1993 *Multiobjective Optimization using the Niche Pareto Genetic Algorithm* IlliGAL Report 93005, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign
- Horn J, Nafpliotis N and Goldberg D E 1994 A niched Pareto genetic algorithm for multiobjective optimization *Proc. 1st IEEE Conf. on Evolutionary Computation (Orlando, FL, June 1994)* (Piscataway, NJ: IEEE) pp 82–7
- Hwang C-L, Lai Y-J and Liu T-Y 1993 A new approach for multiple objective decision making *Comput. Operations Res.* **20** 889–99
- Ishibuchi H and Murata T 1996 Multi-objective genetic local search algorithm *Proc. 3rd IEEE Int. Conf. on Evolutionary Computation* (Piscataway, NJ: IEEE) pp 119–24
- Keeney R and Raiffa H 1976 *Decisions with Multiple Objectives* (New York: Wiley)

- Kita H, Yabumoto Y, Mori N and Nishikawa Y 1996 Multiobjective optimization by means of the thermodynamical genetic algorithm *Proc. 4th Int. Conf. on Parallel Problem Solving from Nature (Berlin, 1996) (Lecture Notes in Computer Science 1141)* ed H-M Voigt, W Ebeling, I Rechenberg and H-P Schwefel (Berlin: Springer) submitted
- Krause M and Nissen V 1995 On using penalty functions and multicriteria optimization techniques in facility layout *Evolutionary Algorithms for Management Applications* ed J Biethahn and V Nissen (Berlin: Springer) pp 153–66
- Kundu S and Kawata S 1996 AI in control system design using a new paradigm for design representation *Proc. 4th Int. Conf. on Artificial Intelligence in Design* (Boston, MA: Kluwer)
- Kundu S, Kawata S and Watanabe A 1996 A multicriteria approach to control system design with genetic algorithms *Int. Federation of Automatic Control (IFAC) '96—13th World Congr. (San Francisco, CA, 1996)* (Kidlington: Pergamon) at press
- Kursawe F 1990 *Evolutionsstrategien für die Vektoroptimierung* Diplomarbeit, Universität Dortmund (in German)
- 1991 A variant of evolution strategies for vector optimization *Parallel Problem Solving from Nature (PPSN) (Lecture Notes in Computer Science 496)* (Berlin: Springer) pp 193–7
- Langdon W B 1995 Evolving data structures using genetic programming *Proc. 6th Int. Conf. on Genetic Algorithms (Pittsburgh, PA, July 1995)* ed L J Eshelman (San Mateo, CA: Morgan Kaufmann) pp 295–302
- 1996 Using data structures within genetic programming *Genetic Programming 1996* ed J R Koza, D E Goldberg, D B Fogel and R L Riolo (Cambridge, MA: MIT Press) pp 141–9
- Li Y, Gen M and Ida K 1996 Evolutionary computation for multicriteria solid transportation problem with fuzzy numbers *Proc. 3rd IEEE Int. Conf. on Evolutionary Computation* (Piscataway, NJ: IEEE) pp 596–601
- Liepins G E, Hilliard M R, Richardson J and Palmer M 1990 Genetic algorithms applications to set covering and traveling salesmen problems *Operations Research and Artificial Intelligence: the Integration of Problem-Solving Strategies* ed D E Brown and C C White (Boston, MA: Kluwer) pp 29–57
- Louis S and Rawlins G J E 1993 Pareto optimality, GA-easiness, and deception *Proc. 5th Int. Conf. on Genetic Algorithms (Urbana-Champaign, IL, July 1993)* ed S Forrest (San Mateo, CA: Morgan Kaufmann) pp 118–23
- Michielssen E and Weile D S 1995 Electromagnetic system design using genetic algorithms *Genetic Algorithms in Engineering and Computer Science* ed G Winter, J Périaux, M Galán and P Cuesto (Chichester: Wiley) ch 18, pp 345–69
- Mori N, Yoshida J, Tamaki H, Kita H and Nishikawa Y 1995 A thermodynamical selection rule for the genetic algorithm *Proc. 2nd IEEE Int. Conf. on Evolutionary Computation (Perth, November–December 1995)* vol 1 (Piscataway, NJ: IEEE) pp 188–92
- Murata T and Ishibuchi H 1995 MOGA: multi-objective genetic algorithms *Proc. 2nd IEEE Int. Conf. on Evolutionary Computation (Perth, November–December 1995)* vol 1 (Piscataway, NJ: IEEE) pp 289–94
- Osyczka A 1984 *Multicriterion Optimization in Engineering (with FORTRAN Programs)* (Chichester: Ellis Horwood)
- Osyczka A and Kundu S 1995 A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm *Structural Optimization* **10** 94–9
- 1996 A modified distance method for multicriteria optimization, using genetic algorithms *Comput. Industrial Eng. J.* **30** at press
- Pareto V 1896 *Cours d'Économie Politique, I* (Lausanne: Rouge)
- Poloni C 1995 Hybrid GA for multi objective aerodynamic shape optimisation *Genetic Algorithms in Engineering and Computer Science* ed G Winter, J Périaux, M Galán and P Cuesto (Chichester: Wiley) pp 397–415
- Richardson J T, Palmer M R, Liepins G and Hilliard M 1989 Some guidelines for genetic algorithms with penalty functions *Proc. 3rd Int. Conf. on Genetic Algorithms (Fairfax, VA, June 1989)* ed J D Schaffer (San Mateo, CA: Morgan Kaufmann) pp 191–7
- Ritzel B J, Eheart W and Ranjithan S 1994 Using genetic algorithms to solve a multiple objective groundwater pollution containment problem *Water Resources Res.* **30** 1589–603
- Savic D A and Walters G A 1995 Genetic operators and constraint handling for pipe network optimization *Evolutionary Computing (AISB Workshop, Sheffield, 1995) (Springer Lecture Notes in Computer Science (LNCS) 993)* ed T C Fogarty (Berlin: Springer) pp 154–65
- Schaffer J D 1984 *Some Experiments in Machine Learning using Vector Evaluated Genetic Algorithms* Doctoral Dissertation, Department of Electrical Engineering, Vanderbilt University, unpublished
- 1985 Multiple objective optimization with vector evaluated genetic algorithms *Proc. 1st Int. Conf. on Genetic Algorithms (Pittsburgh, PA, July 1985)* ed J J Grefenstette (Hillsdale, NJ: Erlbaum) pp 93–100
- Schaffer J D and Grefenstette J J 1985 Multi-objective learning via genetic algorithms *Proc. 9th Int. Joint Conf. on Artificial Intelligence* vol 1, pp 593–5
- Shaw K J and Fleming P J 1996a Initial study of multi-objective genetic algorithms for scheduling the production of chilled ready meals *2nd Int. Mendel Conf. on Genetic Algorithms (Mendel '96) (Brno, 1996)*; Research Report 623, Department of Automatic Control and Systems Engineering, University of Sheffield
- Simpson A R, Dandy G C and Murphy L J 1994 Genetic algorithms compared to other techniques for pipe optimization *J. Water Resources Planning Management* **120** 423–43
- Srinivas N 1994 *Multiobjective Optimization using Nondominated Sorting in Genetic Algorithms* Master's Thesis, Indian Institute of Technology

- Srinivas N and Deb K 1995 Multiobjective optimization using nondominated sorting in genetic algorithms *Evolutionary Comput.* **2** 221–48
- Stanley T J and Mudge T 1995 A parallel genetic algorithm for multiobjective microprocessor design *Proc. 6th Int. Conf. on Genetic Algorithms (Pittsburgh, PA, July 1995)* ed L J Eshelman (San Mateo, CA: Morgan Kaufmann) pp 597–604
- Steuer R E 1986 *Multiple Criteria Optimization: Theory, Computation, and Application* (New York: Wiley)
- Surrey P D, Radcliffe N J and Boyd I D 1995 A multi-objective approach to constrained optimisation of gas supply networks: the COMOGA method *Evolutionary Computing (AISB Workshop, Sheffield, 1995)* (*Springer Lecture Notes in Computer Science (LNCS) 993*) ed T C Fogarty (Berlin: Springer) pp 166–80
- Takada Y, Yamamura M and Kobayashi S 1996 An approach to portfolio selection problems using multi-objective genetic algorithms *Proc. 23rd Symp. on Intelligent Systems* pp 103–8 (in Japanese)
- Tamaki H, Kita H and Kobayashi S 1996 Multi-objective optimization by genetic algorithms: a review *Proc. 3rd IEEE Int. Conf. on Evolutionary Computation* (Piscataway, NJ: IEEE) pp 517–22
- Tamaki H, Mori M, Araki M, Mishima Y and Ogai H 1995 Multicriteria optimization by genetic algorithms: a case of scheduling in hot rolling process *Proc. 3rd Conf. of the Association of Asian-Pacific Operational Research Societies (APORS '94)* ed M Fushimi and K Tone (Singapore: World Scientific) pp 374–81
- Thierens D 1995 *Analysis and Design of Genetic Algorithms* Doctoral Dissertation, Catholic University of Leuven
- Thierens D and Goldberg D E 1994 Elitist recombination: an integrated selection recombination GA *Proc. 1st IEEE Conf. on Evolutionary Computation (Orlando, FL, June 1994)* (Piscataway, NJ: IEEE) pp 508–12
- Tsoi E, Wong K P and Fung C C 1995 Hybrid GA/SA algorithms for evaluating trade-off between economic cost and environmental impact in generation dispatch *Proc. 1st IEEE Int. Conf. on Evolutionary Computation (Orlando, FL, June 1994)* vol 1 (Piscataway, NJ: IEEE) pp 132–7
- Vemuri V R and Cedeño W 1995 A new genetic algorithm for multi-objective optimization in water resource management *Proc. 1st IEEE Int. Conf. on Evolutionary Computation (Orlando, FL, June 1994)* vol 1 (Piscataway, NJ: IEEE) pp 495–500
- Wallace D R, Jakiela M J and Flowers W C 1994 Design search under probabilistic specifications using genetic algorithms *Computer-Aided Design* **28** 405–20
- Wienke D, Lucasius C and 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) *Anal. Chim. Acta* **265** 211–25
- Yoshida K, Yamamura M and Kobayashi S 1996 Generating a set of Pareto optimal decision trees by GA *Proc. 4th Int. Conf. on Soft Computing (1996)* at press