The Good of the Many Outweighs the Good of the One: Evolutionary Multi-Objective Optimization

David W. Corne, University of Reading, UK
Kalyanmoy Deb, Indian Institute of Technology, Kanpur, India
Peter J. Fleming, University of Sheffield, UK
Joshua D. Knowles, Université Libre de Bruxelles, Belgium

Abstract. We dwell in largely non-technical terms on the essential differences between single-objective optimization and multiple-objective optimization. We argue in particular that single-objective approaches to real-world problems are almost invariably simplifications of the real-problem which make many ideal solutions unreachable to the optimization method. We promote the use of multi-objective optimization methods, particularly those arising from the evolutionary computation community. We point out that the state of the art in the field of evolutionary multi-objective optimization is such that fast and effective techniques are now available which are capable of finding a well-distributed set of diverse trade-off solutions, with little or no more computational effort than sophisticated single-objective optimizers would have taken to find a single one. The resulting diversity of ideas available through a multi-objective approach leads both to the problem-solver being furnished with a better understanding of the space of possible solutions, and consequently to a better final solution to the problem at hand. We end by very briefly charting the history of the field and hinting at the range of published applications and ongoing research issues.

1 Introduction

Modern ideas for solving optimization problems, particularly *evolutionary computation* (EC) and its many variants, are making their way with profound success into the realm of real-world use. There is an endlessly growing list of applications which have either been tested with promising results on real-world problems, or describe applications in regular deployment, often producing results greatly improved over the previously employed solution.

The proud march of EC into industrial and commercial life is in large part a result of its underlying efficacy as an optimization tool, but also owes much to its *convenience*. EC imposes no requirement for a problem to be formulated in a particular constraint language, or for the function in question to be differentiable (or even continuous), and does not ask that the number of parameters be limited, or of any particular data-type. Rather, EC is universally applicable (it may not always work, but at least it can always be attempted), and asks only two key things of the practitioner: (1) that there be some (almost any) way to encode a candidate solution to the problem, and (2) that there be an algorithm in place for calculating the quality of any such encoded solution – the so-called *fitness function*.

However, despite freeing the practitioner from the shackles applied by many other optimization techniques, there is one essential freedom which is, as we contend, far too little exploited. Note that the fitness function, the necessity for which is one of very few burdens imposed *a priori* by EC on its user, *need not be restricted to return only a single scalar value*. Many may be forgiven for assuming that the prowess of EC is solely to be found in the optimization of a single objective, since until recently the algorithmic machinery available for handling *multiple* objectives well was not particularly convincing or effective. However, things are now changing markedly in this regard.

As recently well-put in Michalewicz and Fogel (2000), there are basically two ways to solve problems: one is to *simplify* the model so that traditional methods of optimization can be applied, and the other is to keep the model as it is – with no simplifications or transformations which pander to the needs of specific approaches – and use a 'non-traditional' approach, Our observation about recent decades of 'real-world' optimization is that the first of these has been used far more deeply than practitioners are ready to realize. That is, by far the most common way to handle multi-criteria optimization problems has been to *simplify* them as single-criterion problems, and *this is commonly done without conscious reflection*! Evidence for this view is apparent in the frequent appearance of 'penalty-based methods' in which a number of different components of a quality function (such as mass and aerodynamicity, or makespan and flowtime, or cost and reliability, and so on …) are simply combined into a single scalar objective via a weighted, usually linear sum.

It is currently endemic in the optimization community to treat this way of combining criteria as natural and, by implication, harmless. However, we would beg to differ, and would stress the following points. First, transforming a multi-criteria problem into a single-criterion problem is, in all but pathological cases, quite a radical simplification of the problem, so much so that the new optimization problem which results is saliently different, and very often not the problem that really needs to be solved. Second, this simplifying transformation is, arguably, *no longer necessary* in the light of recent progress in the field of evolutionary multi-objective optimization. We now have a sparkling collection of algorithms which can apply to multi-criteria problems without requiring their simplification, and furthermore these algorithms are becoming increasingly fast, efficient, and effective. Certainly, there is much progress still to be made, and we do not yet have a *panacea* for every awkward feature that a particular real-world problem will present. However, our chief point is that the proper treatment of a multi-criterion real-world problem without this simplification is now an eminently viable and recommended approach.

2 On the Multi-Criteria Nature of Real-World Problems

Real-life is rarely characterized by a position on the real number line, and even $\Re^2$ is frequently too impoverished in descriptive ability. Consider the apparently harmless question: "How do you like this article?". If we ask you to rate it with an integer in the range [0 , 10], then (we claim, unless you feel it sings 'zero' from every corner), you will find it quite difficult, and feel compelled to qualify your answer (e.g. "Well, I think it's a 5, but the list of references is better than that suggests, and the attempts at humor much worse, …"). If instead we provide a standard paper review form, which asks for ratings in perhaps ten distinct categories, you will feel that your responses give a much fairer reflection, and may even find it easier to fill out this form than distil your feelings into a single number. A more particular problem with asking for a single-value rating is this: suppose IEEE members have, about a year from now, rated, from 0 (poor) to 10 (excellent) every article that appeared in the first volume of the IEEE Neural Networks Society Newsletter. Suppose further that five of these articles have a mean rating of 9.5. Which of these is the 'best'? Immediately it is clear that the response 'they are equally good' is unlikely to be correct, but the key point is that the single-value ratings do not help with this decision. If instead we had several ratings for each article, each rating for a different criterion, there would be much more information available to help discriminate between them. This would have allowed a rater to say 'well, although I'd give it a rating of 9 out of 10 if I had to stick to one number, this multi-criteria system enables me to point out that the list of references is not really complete, and the overall impact it is likely to have is not quite at the level of other articles I have read'.

Thus characterized, we now have a *vector* of scores for each article, and the special multi-criteria concept of *domination* can come into play. One excellent article which scores (10, 8, 9, 10) respectively on a list of four criteria can be said to *dominate* another article whose scores are, respectively, (9, 8, 9, 10); the first article is at least as good as the second in every respect, and better in one respect (the first attribute). Crucially, however, we also have the concept of *non-domination*. Quite different articles may have the profiles (8,10,8,9) and (10,8,7,10) respectively. Neither of these dominates the other, and we say that they are non-dominated with respect to each other. It turns out that we now have a similar difficulty to the single-objective one in which we may require the 'best' of several articles with the same single-value score. The difficulty now is to choose the so-called 'best' of a set of articles which are all non-dominated with respect to each other.

However, the difficulty is not quite the same, for two increasingly revealing reasons. And to see this, we will move back into the world of optimization and consider that we are interested in, say, the design of a telecommunications network, or a factory production schedule, or a VLSI chip. First, in the single-scalar-value scenario, optimization methods are far more likely to present us with only one, or very few, 'optimized' designs. Even if there are several, these are highly likely to be very similar to each other, so it is essentially a single design which scores better than any other design that the optimizer could find in that trial run. Hence, this difficulty does not often arise in single-objective approaches. Second, when this arises in modern multi-criteria approaches, it is not a *problem* at all, but an *opportunity*. In fact, returning a set of non-dominated solutions is a priceless result of the optimization method, in which most likely there will be several very distinct solutions among the non-dominated set. This collection itself conveys a great deal of useful information about the *space* of possible workable designs to the owner of the problem.

For example, whereas the result of single-objective optimization may be a single design for an Olympic Games athletics schedule which minimizes well the degree to which medal ceremonies interfere with events happening at the same time, this solution may be quite awful from the athletes' viewpoint, forcing, for example, successive heats of events such as the 100m, 200m, and 400m to be too close together for comfort. Especially for athletes involved in more than one of these events, performances will be severely hampered and the world-record toll of this Games would be sharply reduced. However, a proper multi-objective optimizer will give the problem solver a good range of quite distinct alternatives.

See Figure 1 – this shows a hypothetical picture of the space of good solutions in our Olympic Games example. The horizontal axis indicates the number of conflicts (overlaps in time) between athletics events and medal ceremonies, increasing to the right. We clearly aim to minimize this. The vertical axis is meant to indicate the degree to which two tiring events (involving one or more athletes performing in both of them) are uncomfortably close together in time. This increases as we go up, and we clearly want to minimize this too. The red and blue circles represent the non-dominated set of solutions returned by a multi-objective optimizer. In fact we will pretend that these are the best solutions possible (i.e. they are the same non-dominated set we would find if we searched the space of solutions exhaustively), and in this case they would represent what we call the *Pareto front*, and each of them is a *Pareto-optimal* point. The name comes from Vilfredo Pareto, an Italian economist and sociologist, known for, among other things, his ground-breaking work on the application of mathematical ideas in economics. This was crystallized in Pareto (1906), which set out the concept of a Pareto optimum.

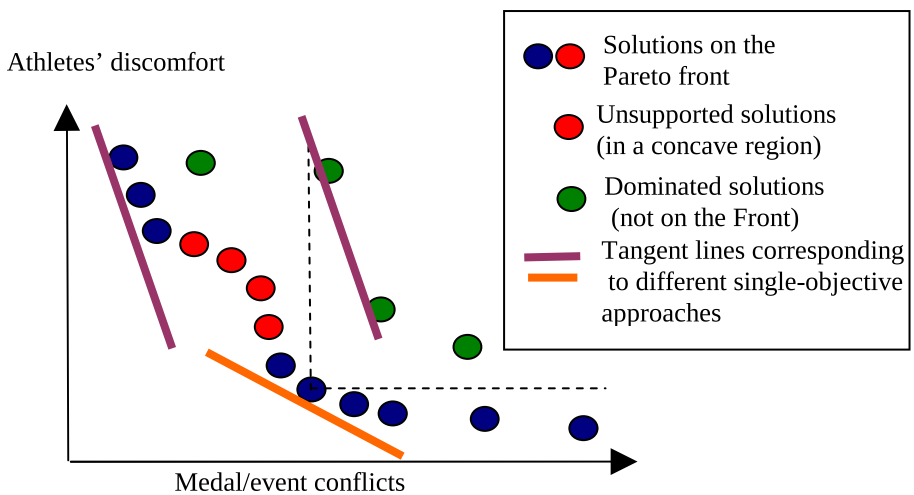

Figure 1: Illustrating salient concepts of Multi-objective optimization on a contrived athletics events scheduling problem.

Now, note the topmost/leftmost blue circle in Figure 1. This corresponds to what we described earlier as the result from our single-objective optimizer. It scores very well at minimizing conflicts between athletics events and medal ceremonies, in fact it is the best possible solution in this sense, however note that it scores very poorly from the atheletes' viewpoint; perhaps, for example, it involves all four successive stages of the 400m in one day.

This single objective optimization approach will probably have been trying to minimize $w_1 m + w_2 d$, where *m* tallies the conflicts on the horizontal axis and *d* represents the degree of discomfort on the vertical axis, and w_1 and w_2 are positive *weights*, or *penalty coefficients*, roughly reflecting the respective importance of these two objectives. Notice that every distinct choice for the penalty coefficients corresponds to a different version of the optimization problem, and correspondingly has a different optimum. In fact, if we say $w_1 m + w_2 d = a$, for some *a*, then we define a specific line with a specific slope, and all points in design space which fall on this line achieve the same single-objective quality. The purple line segments in figure 1 correspond to (parts of) two such lines, for two different values of *a*, for the same set of penalty coefficients. So, with the aid of figure 1 we can see that the optimum value with regard to a particular choice of weights corresponds to the smallest value of *a* for which a line $w_1 m + w_2 d = a$ is tangential to the Pareto front. Whatever choice of weights led to the purple lines, we can see that the extremal point we have discussed (the topmost/leftmost blue circle) is the optimum of that particular single objective function.

What is worth mentioning at this point is that all of the other Pareto solutions in this example are *suboptimal* for this set of weights, So, a simulated annealing approach which happened to find the optimum, for example, will have discarded any of the other Pareto optimal points it may or may not have met on the way. Had we a different set of weights instead, our earlier optimum would now be suboptimal, and a good optimizer would find another point along the Pareto front, corresponding to the tangent which relates to this choice of weights (e.g. perhaps this new set of weights will correspond to the slope of the orange line in figure 1. However, it is also worth noting that *none* of the red points in figure 1 is the optimal solution to any such penalty function. Such 'concave' regions in a Pareto front are very common, and may be very hard for single-objective optimizers to happen upon (and keep), although they may correspond to ideal tradeoffs between the objectives from the problem-solver's viewpoint.

3 Why it Matters

The initial thoughts of an intelligent reader new to multi-objective optimization might go roughly as follows. 'OK, so it is good to have a set of non-dominated points, but maybe these will appear if we simply look at all the points visited by a single optimization run anyway. If not, then we can simply run single-objective optimization many times, but with different penalty coefficients, and thereby get a good collection of points on the Pareto front.'

It turns out that both of these ideas are very over-optimistic. The first is almost a non-starter when we examine it on real problems, and the second, though going in a direction which *does* yield fruit, is hopelessly inefficient in its natural formulation. The point is, having recognized that Pareto fronts are good to have, far more valuable to the problem solver or designer than a single solution to a simplified and *different* problem, the basic algorithmic machinery provided by the portfolio of single-objective optimization research, though rich in capability and ingenuity, is not suited to the task of efficiently finding good approximations to the Pareto front of a multi-objective problem.

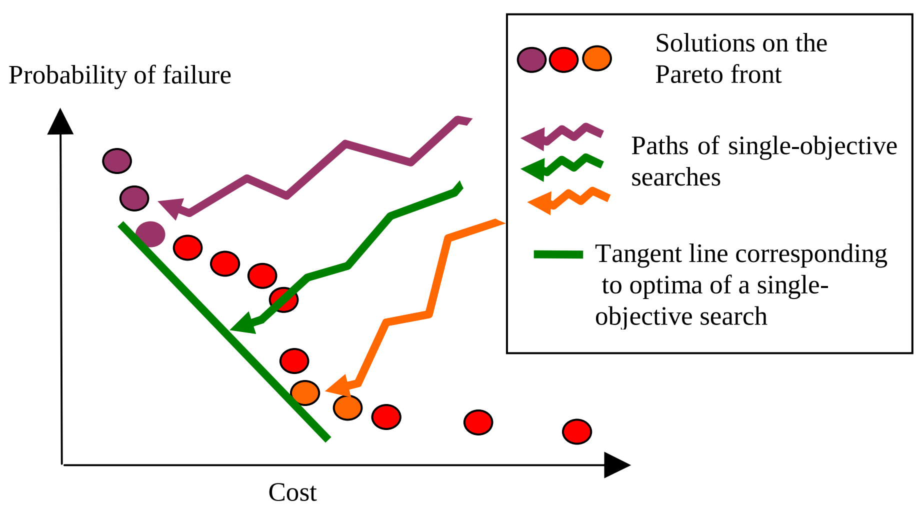

Figure 2: Illustrating, on a generic two-objective problem scenario, the impoverished solution set(s) available to pure single-objective optimization approaches.

Figure 2 helps us reason about this. In Figure 2 we again plot the Pareto optimal points of a hypothetical two-objective optimization problem, in this case labeling the axes with "probability of failure" and "cost", respectively, thus hinting at just part of the immense proportion of real-world problems which can only be properly handled by a multi-objective approach. The meandering arrows each sketch the progress of a single-objective (penalty-function based) approach. The parameters guiding the search of the purple arrow, for example, are likely to find, at best, the purple Pareto optimal points; a similar story applies for the orange arrow. Meanwhile, the green arrow is not necessarily likely to find any Pareto optimal points not found anyway by the purple and orange arrows; this is because the section of the Pareto front which seems to be on its path is concave, and these points are not optima for the 'green' search. Instead, assuming the underlying optimizer is a good one, it will find the points touching the green tangent line. There are many ifs and buts which should really qualify this simple illustration, but the broad message is true. A single search with a penalty function is likely to leave lots of Pareto optimal points undiscovered, and therefore inaccessible to the problem-solver. These include maybe ideal solutions to the problem, but, despite the pedigree of the single-objective method used, remain lost. Repeated single-objective runs each with a different penalty function will, however, have some chance of yielding a picture of the Pareto front, but note that this picture will often be very patchy and sparse. Finding points in concave regions is very much in the lap of the Gods, while the fact that this involves several runs of the single-objective method reveals burdensome time complexity, and this onus is exponentially increased as we raise the number of objectives beyond two, which is very common indeed in real-world applications.

Further, it is instructive to consider the great depth of assumptions inherent in any single-objective penalty-based approach. Without missing the point, let's assume the intention in such an approach is to treat each objective as equally important. We might then imagine such a search would be ideally directed at what happens to be a concave region in Figure 2, since points here are between the extremes of both objectives (and for other reasons, as we already know, hard for single-objective optimization to find in the first place). However, even this simple goal of equal weighting needs *very* careful parameterization of the penalty function. To see this, suppose we expect cost to vary between $1M and $5M, and probability of failure to vary from 0 to 1. For single objective optimization to weight these equally, we would do the equivalent of multiplying the probability objective by around 4M, and expect that to facilitate a search in the direction of the green arrow in Figure 2. However, returning to real life, and in particular the *a priori unknown* nature and position of the Pareto front, the balance of probabilities would have it that such a search will always end up going in entirely the wrong direction and focusing on a single extreme of just one of the objectives. In one problem instance, for example, the Pareto optimal points in Figure 2 may range from 0.9 to 0.95 on the vertical axis, and all the way from $1M to $5M on the horizontal axis. The unwise penalty setting in such a case will have left us stuck in the bottom left corner, finding only high-cost solutions. The main point here is that the success of a penalty-function approach is sensitive to many factors, most of which are completely unknown beforehand. Even the conceptually simple goal we set above, of finding a single-objective function which treats each objective as equally important, is almost never achievable in practice – the situation is of course even worse when we want to coerce a penalty function into expressing more complex goals, which is often the case. However, a 'proper' multi-objective optimization approach simply steps aside from all such sensitivities, naturally treating each objective as an equal. This brings its own issues, but of a much more agreeable nature: no longer need we fear completely *missing* the target; instead, the problem-solver is facilitated by a full map of the target area, and then needs to work further with her advanced multi-objective optimizer to zoom in on the fruitful areas.

To end this section we underline the fact that all this *matters* by considering again the typical structures of real-world problems. It is very rare indeed that a real-world problem has only one objective. For example, the classic traveling salesperson problem (TSP) is the *Drosophila melanogaster* of single-objective optimization research (Lawler et al, 1985), but its real-world counterparts are hardly ever true single-objective problems. If we consider the eponymous case, where the problem is really to find the shortest route between cities for a traveling hawker, the different visits along that route will invariably have different priorities (in terms of likeliness to buy the goods), and an ultimately much longer than optimal route may be greatly favoured since it puts our salesman in Denver at just the right time to buy a lavish lunch for a potentially big buyer. When it does seem to be the case that one objective is all we need, this is often a *cost* measure, which tries to sum up in dollars and cents the total cost of a design. But we all know very well that this is always a gross simplification; the cheaper it is to make our VLSI chip, the more we may have to pay out in the future for replacements and lost business. The more fuel payload we save from a cleverly optimized route for our Saturn probe, the less opportunity we have to steer it away from trouble (and thus save billions of dollars) when it goes off course.

4 The Rise of Evolutionary Multiobjective Optimizers

We must not ignore non-EC approaches, and cite here a selection of classical and recent works in which other optimization paradigms have addressed the needs of multiple objectives (Kung et al, 1975; Benson, 1978; Steuer, 1986;

Dasgupta et al, 1999; Miettinen, 1999). See also Deb (2001) for significant space given to the treatment of the alternative methods in relation to evolutionary computation. In this article, however, we concentrate on EC approaches, (Deb, 2001; Coello et al, 2002), and remark that, as ever, EC-oriented methods certainly seem *broadly* the most effective overall, but that, in particular, the ideal approach to any given complex multiobjective problem will typically be some form of hybrid of an EC method and a classical method. It is worth noting here that the new conference series on Evolutionary Multicriterion Optimization (EMO) is making successful efforts at uniting the various optimization communities which specialize in multi-objective optimization; this is reflected in the first (of what we think will be many) EMO proceedings (Zitzler et al, 2001).

Moving on to discuss the rise of evolutionary multi-objective optimizers, we will first skate through an often-published history. This is to be found in some form in many of the recent theses which have introduced new evolutionary multi-objective algorithms and/or performed extensive case studies (Fonseca, 1995; Van Veldhuizen, 1999; Andersson, 2001; Zitzler, 2001; Knowles, 2002) – there are many more beyond this selection, and Carlos Coello is maintaining a good repository for such references as well as other multi-objective-related resources (software, test problems and so forth) at http://www.lania.mx/~ccoello/EMOO/. The tale goes as follows; after earlier work which seriously considered multi-objective problems but used penalty-function based approaches (Box, 1957; Fogel et al, 1966) serious effort to design *specialist* multi-objective optimization techniques in the evolutionary computation community seem to have started with Schaffer (1984). Schaffer's idea, in the so-called VEGA method, involved considering each objective in turn in different cycles of the optimization. It was quite a while later (following a discussion in Goldberg (1989)) that researchers started to use the notion of *dominance* (see section 2) as a feature of their algorithms. That is, instead of the central scheme of selection (choosing something from the population which will then be subjected to mutation or crossover) being biased by a single-objective value, researchers began, after 1989, to use selection schemes which were based on dominance measures. I.e. the more a candidate solution in the population dominates others, the more likely it is to be selected to progress the optimization. It also seemed crucial to combine this with some form of 'niching' scheme, which ensured that selection was not unduly biased towards fit regions simply because lots of similar solutions were in that region. In such situations, near-copies of a very good individual tend to 'steal' the available chances for selection without themselves adding anything 'new' to the search. Such schemes were attracting much attention in single-objective evolutionary optimization at the time, and were carried over directly into the emerging Evolutionary multiobjective algorithms, where they seemed very helpful in leading to a well-distributed and extensive Pareto tradeoff surface. The early-90s crop of techniques which incorporated such notions (and later versions are still used), prominently included NSGA (Srinivas & Deb, 1994), NPGA (Horn et al, 1994), and MOGA (Fonseca & Fleming, 1993).

For a while after that, progress was not spectacular until it became generally noticed that *elitism*, while not a good thing in single objective optimization, tends to be very beneficial in multi-objective optimization. In the single-objective case, elitism – which essentially means mostly selecting the best so far, tends to quickly lead to premature convergence to suboptimal solutions. However, the corresponding concept in multi-objective optimization is to choose mainly from the current approximation to the Pareto front; i.e. the set of non-dominated solutions in the current population; since there is still a great deal of diversity present in such an 'elite' set (assuming the underlying optimizer is good), good progress towards even better solutions is not hampered, and is usually accelerated. So-called elitist evolutionary multiobjective optimization algorithms started coming to the fore with SPEA (Zitzler & Thiele, 1998; Zitzler et al, 2000), an application by Parks and Miller (Parks & Miller, 1998), ERMOCS (Neef et al, 1999), NSGA-II (Deb et al, 2000), PAES (Knowles & Corne, 2000), PESA (Corne et al, 2000), and many more. Finally, of rampant recent interest are hybrid approaches which combine evolutionary multi-objective optimization and local search in strategically suitable ways. To provide just a snippet of a growing thread, we cite Ishibuchi & Murata (1996) as a seminal example of such work with their MOGLS algorithm, which has been recently re-engineered by Jaszkiewicz (2002) with extremely competitive results.

This post-1995 explosion of interest in new techniques for evolutionary multiobjective optimization was fuelled by the following common experience: on real and complex problems, the new methods are able to evolve good Pareto fronts with no significant extra time cost above that of a single objective optimizer finding just one solution on the front. Hence, treating a multi-objective real world problem as it really should be treated – i.e. without simplifying and changing it to a single-objective pseudo-real problem – was clearly becoming a viable, fast and effective alternative. In recent years, the fruits of this are apparent in a multifarious collection of real-world applications. For example, the civil engineering enterprise of designing and controlling water networks is gaining much from the ascent of this field (Halhal et al, 1997; Chen & Chang, 1998; Reed et al, 2001). The rise of effective multiobjective optimization is also making great impact in engineering design and control (Chipperfield & Fleming, 1996; Dakev et al, 1997; Kim & Ghaboussi, 1999), in various design applications (Park & Grierson, 1999; Parmee et al, 2000; Balling, 2001), for optimization of networks of various kinds (Lo & Chang, 2000; Knowles et al, 2000; Ramírez-Rosado & Bernal-Agustín, 2001), and many other applications far too numerous to list in anything smaller than a large book. Again, the reader interested in getting a good feel for the breadth of work currently going on can refer to the books and the web location previously mentioned, as well as, of course, the standard WWW search engines.

5 Emerging Trends and Concluding Remarks

And so, now, evolutionary multi-objective optimization is ready and able to take on real-world problems in their 'real' form – that is, as multi-objective problems rather than as potentially misleadingly simplified single-criterion formulations. But the field is very much *beginning*. As the efficacy of the various new algorithms has attracted more and more researchers and practitioners to the area, various issues and directions needing further research have become identified. Among these are considerations of the number of objectives in the problem, handling constraints, theoretical support for algorithm design and tailoring, each of which we briefly consider below, but this is just a small collection of topics from a field bubbling with unanswered questions.

Most algorithms are developed and tested with small numbers of objectives (usually 2 or 3) in mind, but it turns out that real-world problems can often have several more; a 10-objective problem is not unusual, for example. There are many issues surrounding this, ranging from the efficacy of our existing algorithms in such circumstances, the need to better understand the landscapes of multi-objective problems, the enterprise of designing algorithms which specialize in many-objective problems, and also the *careful* transformation of such problems into fewer-objective ones. Progress is more apparent so far in the latter line of work, with researchers relying on identified preference relationships between the objectives, either to help the algorithm strategically consider the more important objectives when it needs to, or to exploit the problem solver herself in an interactive system (Shaw & Fleming, 1997; Kita et al, 1999; Barbosa & Barreto, 2001; Duenas & Mortt, 2001; Cvetkovic & Parmee, 2002a; 2002b).

Another issue which is receiving increased attention is that of handling constraints over the decision variables (Binh & Korn, 1997; Surry & Radcliffe, 1997; Fonseca & Fleming, 1998). Dealing with constraints *per se* is difficult, and we can generally approach this in multi-objective optimization the same sort of way we would do so in single-objective optimization. However, the multi-objective framework provides the opportunity for new treatments, in which constraint violation is treated in any of several interesting ways currently under investigation (for example, as another objective to be minimized) (Coello Coello, 2000; Osyczka & Krenich, 2000; Hughes, 2001; Kumar & Ranjithan, 2002).

Theoretical questions are also being asked, and some are being answered, concerning, for example, under what conditions algorithms will converge (Rudolph, 1998; Rudolph & Agapie, 2000), which schemes will ensure convergence at the same time as encouraging diversity (Knowles & Corne, 2003), and which schemes will, at least, converge to the nondominated subset of the candidate solutions encountered during an optimization run (Laumanns et al, 2001). The consequences and side issues of the No Free Lunch theorem (Wolpert & Macready, 1997) for multi-objective optimization are also beginning to be addressed (Corne & Knowles, 2002/3). Next, towards tailoring and tuning multi-objective optimizers, and following similar work only just starting to bear fruit in the single-objective optimization community (Merz, 2000; Merz & Freisleben, 2000), some researchers are investigating whether we can understand the structure of multi-objective landscapes well enough to tune the features of hybrid evolutionary and local search approaches (Knowles & Corne, 2002; Ishibuchi & Yoshida, 2002), since such approaches are among the most spectacularly well-performing in the field on certain problems (Ishibuchi & Murata, 1996; Czyzak & Jaszkiewicz, 1997; Jaszkiewicz, 2002). Finally, many researchers are bringing the more recent ideas fledged in single-objective optimization into the multi-objective realm, with notable promise shown by, for example, particle-swarm optimization based multi-objective search (Coello Coello & Lechuga, 2002; Hu & Eberhart, 2002; Parsopoulos & Vrahatis, 2002), and similar for ant colony optimization (Mariano & Morales, 1999; Gravel et al, 2001) and differential evolution (Abbass et al, 2001).

To find out more about various aspects of this fast maturing research area, readers can turn to a multitude of resources available on the WWW, via which can be found bibliographies, online articles, public domain code for many of the top-drawer algorithms, and information on relevant conferences and other events. We have already mentioned the site: http://www.lania.mx/~ccoello/EMOO/, which provides pointers to all such resources. We would additionally mention the link: http://conferences.ptrede.com/emo03, which contains information about the upcoming

Evolutionary Multicriterion Optimization conference.

In conclusion, we aim in this article to bring attention to the rising success of the 'truly' multi-objective approach to solving real-world problems. That is, an approach where the algorithm is engineered towards finding an approximation to the Pareto front of a problem, rather than specialized towards optimizing single objectives. Such methods arising from the evolutionary computation community have been our focus, and the field of evolutionary multi-objective optimization is now rampant with capable algorithms, successful applications, and hot research questions. Perhaps the main message is that it is no longer necessary to automatically *simplify* a real-world problem by transforming it into one objective. We contend this is often unwise, and unlikely to be necessary. The various techniques we have discussed are now readily available for treating a problem in its realistic multi-objective form. We have the technology, and solicit its deployment, since we will understand it better the more it gets used.

References

Abbass, H.A., Sarkar, R., Newton, C. (2001). PDE: A Pareto-frontier Differential Evolution Approach for Multi-objective Optimization Problems, I *Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001)*, Vol. 2, IEEE Service Center, Piscataway, New Jersey, pp. 971—978.

Andersson, J. (2001) *Multiobjective Optimization in Engineering Design*, PhD Thesis, Division of Fluid and Mechanical Engineering Systems, Linköpings University, Sweden.

Balling, R.J. (2001) Pareto sets in decision-based design, *Journal of Engineering Valuation and Cost Analysis*, **3**: 189—198.

Barbosa, H.J.C., Barreto, A.M.S. (2001) An interactive genetic algorithm with co-evolution of weights for multiobjective problems, in L. Spector et al. *Proceedings of the Genetic and Evolutionary Computation Conference* (GECCO-2001), pp. 203–210, Morgan Kaufmann Publishers, San Francisco, CA.

Benson, H.P. (1978) Existence of efficient solutions for vector maximization problems, *Journal of Optimization Theory and Applications*, **26**(4): 569–580.

Binh, T.T., Korn, U. (1997) Multiobjective Evolution Strategy for Constrained Optimization Problems, In *Proc. of the IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics*, pages 357–362, Berlin, Germany.

Box, G.E.P. (1957) Evolutionary operation: method for increasing industrial productivity, *Applied Statistics*, **6**(2): 81–101.

Chen, H.W., Chang, N–B. (1998) Water pollution control in the river basin by fuzzy genetic algorithm based multiobjective programming modeling, *Water Science & Technology*, **37**(8): 55–63.

Chipperfield, A.J. and Fleming, P.J. (1996) Multiobjective Gas Turbine Engine Controller Design Using Genetic Algrithms. *IEEE Transactions on Industrial Electronics*, **43**(5).

Coello Coello, C.A. (2000) Treating Constraints as Objectives for Single-Objective Evolutionary Optimization, Engineering Optimization, **32**(3): 275–308.

Coello Coello, C.A., Lechuga, M.S. (2002) MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization, in *Congress on Evolutionary Computation (CEC'2002)*, Vol. 2, pp. 1051--1056, IEEE Service Center, Piscataway, New Jersey.

Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B. (2002) *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer, Boston.

Corne, D.W., Knowles, J.D. (2003) No Free Lunch and Free Leftovers Theorems for Multiobjective Optimization Problems". Evolutionary *Multi-Criterion Optimization (EMO 2003) Second International Conference*, Faro, Portugal, April 2003, Proceedings, pp. 327-341, Springer LNCS.

Corne, D.W., Knowles, J.D., Oates, M.J. (2000) The Pareto envelope-based selection algorithm for multiobjective optimisation, in *Proceedings of Parallel Problem Solving from Nature VI (PPSN-VI)*, Springer, pp. 839–848.

Cvetkovic, D., Parmee, I.C. (2002a) Preferences and their Application in Evolutionary Multiobjective Optimisation. *IEEE Transactions on Evolutionary Computation*, **6**(1): 42—57.

Cvetkovic, D., Parmee, I.C. (2002b) Agent-based Support within an Interactive Evolutionary Design System, in I.C. Parmee (Editor), *Proceedings of the Fifth International Conference on Adaptive Computing Design and Manufacture (ACDM 2002)*, pp. 355–367, Springer-Verlag.

Czyzak, P., Jaszkiewicz, A. (1997) Pareto Simulated Annealing**,** In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making. Proceedings of the XIIth International Conference*, pages 297–307, Hagen, Germany, Springer-Verlag.

Dakev, N.V., Whidborne, J.F., Chipperfield, A.J, Fleming, P.J. (1997) H_∞ design of an EMS control system for a maglev vehicle using evolutionary algorithms. *Proceedings of IMechE-I Part I: Journal of Systems and Control Engineering*, **311**(4): 345—355.

Dasgupta, P., Chakrabarti, P.P., DeSarkar, S.C. (1999) *Multiobjective Heuristic Search*, Vieweg, Wiesbaden, Germany.

Deb, K. (2001) *Multi-Objective Optimization using Evolutionary Algorithms*, Wiley, Chichester, UK.

Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. (2000) A fast elitist nondominated sorting genetic algorithm for multi-objective optimization – NSGA-II, in *Proceedings of Parallel Problem Solving from Nature VI (PPSN-VI)*, Springer, pp. 849–858.

Duenas, A., Mort, N. (2001) A Genetic Algorithm for Multiobjective Optimisation using the interactive Sequential Multiobjective Problem Solving Method, *Proceedings of the International Conference on Artificial Intelligence (IC-AI'2001)*, pp. 633–639, CSREA Press, Las Vegas, Nevada, June 2001.

Fogel, L.J., Owens, A.J., Walsh, M.J. (1966) *Artificial Intelligence through Simulated Evolution*, Wiley, New York.

Fonseca, C.M. (1995) *Multiobjective Genetic Algorithms with Applications to Control Engineering problems*, PhD thesis, Dept of Automatic Control and Engineering, University of Sheffield, UK.

Fonseca, C.M., Fleming, P.J. (1993) Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization, in *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Francisco, pp. 416–423.

Fonseca, C.M., Fleming, P.J. (1998) Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms--Part I: A Unified Formulation. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, **28**(1): 26–37.

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

Gravel, M.,Price, W.L., Gagné, C. (2001). *Scheduling Continuous Casting of Aluminum Using a Multiple-Objective Ant Colony Optimization Metaheuristic*. Technical Report 2001-004, Faculté des Sciences de L'Administration, Université Laval, Québec, Canada.

Halhal, D., Walters, G.A., Ouazar, D., Savic, D.A. (1997) Multi-objective improvement of water distribution systems using a structured messy genetic algorithm approach, *Journal of Water Resources Planning & Management ASCE*, **123**(3): 137–146.

Horn, J., Napfliotis, N., Goldberg, D.E. (1994) A niched Pareto genetic algorithm for multiobjective optimization, in *Proceedings of the first IEEE Conference on Evolutionary Computation*, pp. 82–87.

Hu, X., Eberhart, R. (2002) Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization, in *Congress on Evolutionary Computation (CEC'2002)*, Vol. 2, pp. 1677–1681, IEEE Service Center, Piscataway, New Jersey.

Hughes, E.J. (2001). Constraint Handling With Uncertain and Noisy Multi-Objective Evolution, *Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001)*, Vol. 2, IEEE Service Center, Piscataway, New Jersey, pp. 963–970.

Ishibuchi, H. and Murata, T. (1996) Multi-objectie genetic local search algorithm, in *Proc. Of the 3rd IEEE International Conference on Evolutionary Computation*, pp. 119–124, IEEE Press.

Ishibuchi, H., Yoshida, T. (2002) Hybrid Evolutionary Multi-Objective Algorithms, in A. Abraham, J. Ruiz-del-Solar, M. Köppen (eds.), *Soft Computing Systems: Design, Management and Applications*, IOS Press, pp. 163–172.

Jaszkiewicz, A. (2002) Genetic local search for multi-objective combinatorial optimization, *EJOR*, **137**(3): 50–71.

Kim, Y–J., Ghaboussi, J. (1999) A New Method of Reduced Order Feedback Control Using Genetic Algorithms. *Earthquake Engineering and Structural Dynamics*, **28**(2): 193-212.

Kita, H., Shibuya, M., Kobayashi, S. (1999) Integration of multi-objective and interactive genetic algorithms and its application to animation design, In *Proceedings of IEEE Systems, Man, and Cybernetics*, volume III, pages 646–651.

Kumar, S.V., Ranjithan, S. (2002) Evaluation of the Constraint Method-Based Evolutionary Algorithm (CMEA) for a Tree-Objective Optimization Problem, *GECCO-2002 Proceedings of the Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers, New York.

Kung, H.T., Luccio, F. and Preparata, F.P. (1975) On finding the maxima of a set of vectors, *Journal of the Association for Computing Machinery*, **22**(4): 469–476.

Knowles, J.D. (2002) *Local Search and Hybrid Evolutionary Algorithms for Pareto Optimization*, PhD Thesis, Department of Computer Science, University of Reading, UK.

Knowles, J.D., Corne, D.W. (2000) Approximating the non-dominated front using the Pareto Archived Evolution Strategy, *Evolutionary Computation*, **8**(2): 149–172.

Knowles, J.D., Corne, D.W. (2002) Towards landscape analyes to inform the design of a hybrid local search for the multiobjective quadratic assignment problem, in A. Abraham, J. Ruiz-del-Solar, M. Köppen (eds.), *Soft Computing Systems: Design, Management and Applications*, IOS Press, pp. 271–279.

Knowles, J.D., Corne, D.W. (2003) Knowles, J.D. and Corne, D.W. (2003) Properties of an Adaptive Archiving Algorithm for Storing Nondominated Vectors. *IEEE Transactions on Evolutionary Computation*, **7**(2), pp. 100-116.

Knowles, J.D., Oates, M.J., Corne, D.W. (2000) Multiobjective Evolutionary Algorithms Applied to two Problems in Telecommunications, *BT Technology Journal*, **18**(4):51-64.

Laumanns, M., Thiele, L., Deb, K., Zitzler, E. (2001) *On the Convergence and Diversity-Preservation Properties of Multi-Objective Evolutionary Algorithms*, Technical Report 108, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, May 2001.

Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G. and Shmoys, D.B. (eds.) (1995) *The Travelling Salesman Problem: A Guided Tour of Combinatorial Optimization*, Wiley, Chichester, UK.

Lo, C–C., Chang, W–H. (2000) A Multiobjective Hybrid Genetic Algorithm for Capacitated Multipoint Network Design Problem. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, **30**(3):461—470.

Mariano, C.E., Morales, E. (1999) *A Multiple Objective Ant-Q Algorithm for the Design of Water Distribution Irrigation Networks*, Technical Report HC-9904, Instituto Mexicano de Tecnologıa del Agua, June.

Merz, P. (2000) *Memetic Algorithms for Combinatorial Optimization Problems: Fitness Landscapes and Effective Search Strategies*, Ph.D. thesis, Department of Electrical Engineering and Computer Science, University of Siegen, Germany, 2000.

Merz, P., Freisleben, B. (2000) Fitness Landscape Analysis and Memetic Algorithms for the Quadratic Assignment Problem, *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 4, pp. 337–352.

Michalewicz, Z. and Fogel, D.B. (2000) *How to solve it: Modern Heuristics*, Springer Verlag, Berlin.

Miettinen, K. (1999) *Nonlinear Multiobjective Optimization*, Kluwer, Boston.

Neef, M., Thierens, D., Arciszewski, H. (1999) A case study of a multiobjective recombinative genetic algorithm with coevolutionary sharing, in *Proceedings of the 1999 IEEE Congress on Evolutionary Computation (CEC'99)*, pp. 796–803.

Osyczka, A., Krenich, S. (2000) A New Constraint Tournament Selection Method for Multicriteria Optimization using Genetic Algorithm**,** In *2000 Congress on Evolutionary Computation*, volume 1, pages 501–507, Piscataway, New Jersey, July 2000. IEEE Service Center.

Pareto, V. (1906) *Manuale di economica politica*.

Parmee, I.C., Cvetkovic, D., Watson, A.H., Bonham, C.R. (2000) Multi-objective Satisfaction within an Interactive Evolutionary Design Environment. *Evolutionary Computation*, **8**(2): 197—222.

Park, K–W., Grierson, D.E. (1999) Pareto-Optimal Conceptual Design of the Structural Layout of Buildings Using a Multicriteria Genetic Algorithm. *Computer-Aided Civil and Infrastructure Engineering*, **14**(3):163—170.

Parks, G.T., and Miller, I. (1998). Selective Breeding in a Multiobjective Genetic Algorithm**,** In A. E. Eiben, M. Schoenauer, and H.-P. Schwefel, editors, *Parallel Problem Solving From Nature -- PPSN V*, pp. 250–259, Amsterdam, Holland. Springer-Verlag.

Parsopoulos, K.E., Vrahatis, M.N. (2002) Particle Swarm Optimization Method in Multiobjective Problems, *Proceedings of the 2002 ACM Symposium on Applied Computing (SAC 2002)*, pp. 603–607.

Ramírez-Rosado, I–L., Bernal-Agustín, J–L. (2001) Relability and Cost Optimization for Distribution Networks Expansion Using an Evolutionary Algorithm, *IEEE Transactions on Power Systems*, **16**(1)111—118.

Reed, P.M., Minsker, B.S., Goldber, D.E. (2001) A multiobjective approach to cost effective long-term groundwater monitoring using an elitist nondominated sorted genetic algorithm with historical data, *Journal of Hydroinformatics*, **3**(2):71—89.

Rudolph, G. (1998). Evolutionary Search for Minimal Elements in Partially Ordered Finite Sets. In V.W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben, editors, *Evolutionary Programming VII, Proceedings of the 7th Annual Conference on Evolutionary Programming*, pages 345-353, Berlin, 1998. Springer .

Rudolph, G., Agapie, A. (2000) Convergence Properties of Some Multi-Objective Evolutionary Algorithms, In *Proceedings of the 2000 Conference on Evolutionary Computation*, volume 2, pages 1010-1016, Piscataway, New Jersey, July 2000. IEEE Press.

Schaffer, J.D. (1984) *Some Experiments in Machine Learning using Vector Evaluated Genetic Algorithms*, PhD Thesis, Vanderbilt University, Nashville, TN.

Shaw, K.J., Fleming, P.J. (1997) Use of Rules and Preferences for Schedule Builders in Genetic Algorithms Production Scheduling**,** in D. Corne and J. Shapiro, editors, *Proceedings of the AISB'97 Workshop on Evolutionary Computation. Lecture Notes in Computer Science No. 1305*, Manchester University, 1997. Springer-Verlag.

Srinivas, N., Deb, K. (1994) Multiobjective function optimization using non-dominated sorting genetic algorithm, *Evolutionary Computation*, **2**(3): 221–248.

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

Surry, P.D., Radcliffe, N.J. (1997) The COMOGA Method: Constrained Optimisation by Multiobjective Genetic Algorithms. *Control and Cybernetics*, **26**(3): 391–412.

Van Veldhuizen, D.A. (1999) *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*, PhD Thesis, Dept of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Paterson Air Force Base,Ohio.

Wolpert, D.H. and Macready, W.G. (1997) No Free Lunch Theorems for Optimization, *IEEE Transactions on Evolutionary Computation* **1**(1): 67–82.

Zitzler, E. (2001) *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, PhD Thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.

Zitzler, E., Thiele, L. (2000). Multiobjective optimization using evolutionary algorithms – a comparative case study, in *Proceedings of Parallel Problem Solving from Nature V (PPSN-V)*, Springer, pp. 292–301.

Zitzler, E., Deb, K., Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results, *Evolutionary Computation*, **8**(2): 145–148.

Zitzler, E., Deb, K., Thiele, L., Coello, C., Corne, D.W. (eds.) (2001) *Evlutionary Multi-Criterion Optimization: First International Conference*, Springer LNCS 1993, Springer, 714pp.