School of Engineering, Computer Science
and Mathematics,
The University of Exeter. EX4 4QF. UK.
http://www.ex.ac.uk/secam

Simulated Annealing and Greedy Search for Multi-objective Optimisation

Richard M. Everson, Jonathan E. Fielsend, Kevin I. Smith

May 2009

Abstract

Simulated annealing generalises greedy or elitist search methods by permitting states which are not an improvement over previous solutions, and for single-objective problems these exploratory moves permit escape from local minima. We examine two classes of multi-objective simulated annealing algorithm: those in which a single solution comprises the state and those in which a set of mutually non-dominating solutions form the state. We compare ways of determining the 'energy' of the state in simulated annealing, including the dominated volume and we relate the greedy versions of these annealers (zero computational temperature) to well-established multi-objective optimisation algorithms.

Empirical tests on the DTLZ test problems show that (a) a single solution state is often more efficient than a set-based state; and (b) that exploratory algorithms are out-performed by their greedy counterparts. These results lead to an examination of the róle of local fronts. Problems for which the Pareto front cannot be located via a sequence of infinitesimal perturbations are defined to be non-infinitesimally greedy searchable. We present examples of non-infinitesimally greedy searchable test problems, which are nonetheless searchable using finite perturbations, but which may be adjusted to be arbitrarily hard even with large finite perturbations.

Keywords: Simulated annealing, greedy search, multi-objective, optimisation, test problems.

1 Introduction

In recent years it has been recognised that many, if not all, practical optimisation problems involve the optimisation of more than a single objective. This recognition has been both inspired by and has prompted the development of a variety of practical techniques for locating the optimum, which for multi-objective problems generally comprises a set of more than one, and possibly infinitely many, solutions. Elements of this set—the Pareto optimal set—cannot be ordered because while one may be better than another for a particular objective it is worse on at least one other objective; formally, members of the Pareto set are mutually *non-dominating*.

The majority of techniques for locating approximations to the Pareto front may be categorised as evolutionary algorithms, and include a variety of genetic algorithms (e.g. Deb et al., 2002a; Zitzler and Thiele, 1999; Fonseca and Fleming, 1995) and evolution strategies (ES, e.g. Knowles and Corne, 1999; Laumanns et al., 2002; Fieldsend et al., 2003). Evolutionary algorithms usually maintain a population of putative solutions to the optimisation problem, which allows them to be adapted to the multi-objective case in which a set of solutions is sought. Simulated annealing (SA) (Kirkpatrick et al., 1983), a well-known algorithm for solving single-objective optimisation problems, however, has received relatively little attention from the multi-objective optimisation community. Given a current approximate solution to the optimum, the optimiser *state,* a *greedy* or *elitist* search adopts as the new approximation a perturbation to the current solution that improves the objective. SA on the other hand may be regarded as a greedy search in which non-greedy perturbations (that is, those that degrade the objective) are permitted because these *exploratory* perturbations may lead to eventual improvements in the objective.

The magnitude of permitted exploratory perturbations, measured as the difference in *energy* between the current and perturbed solution, is decreased during the optimisation by decreasing a computational temperature according to an annealing schedule; in the low temperature limit the search becomes entirely greedy. A particular attraction of SA is the existence of Geman & Geman's (1984) proof that guarantees convergence to the global minimum for single objective problems provided that the annealing rate is sufficiently gradual.
Most schemes that adapt simulated annealing to more than one objective (Serafini, 1994; Ulungu et al., 1999; Czyżak and Jaszkiewicz, 1998; Nam and Park, 2000; Hapke et al., 2000; Suppapitnarm et al., 2000; Tuyttens et al., 2003; Bandyopadhyay et al., 2008) have focused on a *single* solution, and determine the quality of perturbations to that single solution using a weighted sum of objectives or an archive of the best mutually non-dominating solutions discovered thus far in the optimisation. The multi-objective simulated annealing, MOSA, scheme proposed by Smith et al. (2008) is, at least, competitive with some well known MOEAs (Deb et al., 2002a; Zitzler and Thiele, 1999; Fonseca and Fleming, 1995) on standard test problems (Deb et al., 2001, 2002b).
An alternative to focusing on a single solution is to regard the mutually non-dominating *set* of solutions discovered thus far as the subject of the annealing. Algorithms based on this idea have not been fully explored in the multi-objective simulating annealing literature, although Fleischer (2003) has proposed a simulated annealer based on dominated volume. In light of the close connection with evolutionary schemes, which may be regarded as set-based multi-objective greedy searchers or set-based simulated annealers at temperature zero, this paper explores the connections and relative efficiencies of set-based and single-solution-based simulated annealing schemes together with their zero temperature or greedy counterparts. The energy difference between two solutions is central to simulated annealing and we therefore explore alternative formulations of the energy change.
Following some introductory material, the components of a general search algorithm are discussed in the context of a general multi-objective simulated annealing algorithm. Well known algorithms are shown to be specializations of this general algorithm to greedy searching, depending on the choice of energy and whether a single solution or a set of solutions is regarded as the current state. Subsequently, in section 3 single solution state and set-based state simulated annealing algorithms are described, together with their greedy counterparts. These algorithms are empirically compared on standard test problems in section 4. Somewhat surprisingly, it is found that the greedy versions of the algorithms, which exclude exploratory perturbations, are generally more efficient than versions which permit exploratory moves. This suggests that, in contrast to single objective problems, the optimum of many multi-objective problems can be located by a greedy algorithm. Reasons for the success of greedy algorithms and the fitness landscape of extant test problems, including those with many local fronts, are explored and new test problems, which require exploratory perturbations to locate the optimum, are described in section 5.

The goal of multi-objective optimisation is to simultaneously maximise or minimise D objectives, y_i, which are functions of P decision variables, $\mathbf{x} = (x_1, x_2, \ldots, x_P) \in X$. Without loss of generality, it may be assumed that the objectives are to be minimised, so that the multi-objective optimisation problem may be expressed as:

$$\text{Minimise} \quad \mathbf{y} = \mathbf{f}(\mathbf{x}) \equiv (f_1(\mathbf{x}), \ldots, f_D(\mathbf{x})) \tag{1}$$

where the decision variables may also be subject to constraints.
Dominance is generally used to compare two multi-objective quantities $\mathbf{y}$ and $\mathbf{y}'$. If $\mathbf{y}$ is no worse for all objectives than $\mathbf{y}'$ and wholly better for at least one objective it is said that $\mathbf{y}$ *dominates* $\mathbf{y}'$, written $\mathbf{y} \prec \mathbf{y}'$. Domination in objective space is usually extended to decision space: thus if $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}')$ it is said that $\mathbf{x}$ dominates $\mathbf{x}'$: $\mathbf{x} \prec \mathbf{x}'$.
The dominates relation is not a total order and two solutions are said to be *mutually non-dominating* if neither dominates the other. A set is said to be a *non-dominating set* if no element of the set dominates any other. In this paper the following notation is used to denote the operation of forming a mutually non-dominating set from a general set A by removing any element of A that is dominated by another element:

$$\text{nondom}(A) \stackrel{\text{def}}{=} \{\mathbf{x} \in A : \mathbf{x} \not\prec \mathbf{x}' \text{ for } \mathbf{x}' \in A\}. \tag{2}$$

A solution $\mathbf{x}$ is said to be *globally non-dominated*, or *Pareto-optimal*, if no other feasible solution dominates it in the objective space. The set of all Pareto-optimal solutions is known as the Pareto set and

Algorithm 1 Multi-objective simulated annealing

Inputs: $\{L_k\}_{k=1}^K$ *Sequence of epoch durations*
$\{T_k\}_{k=1}^K$ *Sequence of temperatures,* $T_{k+1} < T_k$
ω *Initial state*

```
1:  F := ω                                                  Initialise archive
2:  for k := 1, ..., K
3:      for i := 1, ..., L_k
4:          ω' := perturb(ω)
5:          δE(ω, ω') := energyChange(ω, ω')
6:          u := rand(0, 1)
7:          if u < min(1, exp(−δE(ω, ω')/T_k))
8:              ω := ω'                                     Accept new current state
9:              foreach y ∈ ω'                              Maintain the archive
10:                 if ¬(∃z ∈ F s.t. z ≺ y)
11:                     F := {z ∈ F : y ⊀ z}                Remove dominated points from F
12:                     F := F ∪ {y}                        Add y to F
```

the image in objective space of the Pareto set under $\mathbf{f}$ is known as the Pareto-optimal front, $\mathcal{P}$; solutions in the Pareto set represent all possible optimal trade-offs between competing objectives. When using heuristic algorithms, the non-dominated set produced by one, or several, runs will only be an approximation to the true Pareto front; some care with terminology is therefore required, and in this paper the set produced by such an algorithm is referred to as the archive of the estimated Pareto set, denoted by F.

2 Search Techniques

Many optimisation procedures can be decomposed into the following components: a *state*, denoted by ω, which is a set of one or more solutions $\mathbf{x}$; a method of generating a new or *perturbed* state ω' from ω; an *energy* of a state, which in single-objective optimisation, is the objective to be minimised, together with a method of comparing the quality of two states, *energy change*, $\delta E(\omega, \omega')$; and finally a rule to determine whether to accept the perturbed state as the new current state based on $\delta E(\omega, \omega')$. The (negative) energy is often termed the *fitness*, while the perturbation to a solution is equivalent to a *mutation*. A search then proceeds iteratively from an initial state by perturbing the state and determining whether the energy of the perturbed state is lower (fitter) than the original; perturbed states with lower energies are generally adopted as the new state, but some algorithms, particularly simulated annealing, permit *exploratory* moves in which a state with a higher energy is accepted. Having accepted the new state as the current state, the cycle repeats until some convergence criterion is met. Since the state need not contain the best solution visited so far and for multi-objective problems there are usually several mutually non-dominating optimal solutions, many algorithms maintain an additional *archive* or record of the optimal solutions visited thus far in the search.

As will become apparent, many optimisation procedures may be regarded as particular cases of a general multi-objective simulated annealing algorithm, which is summarised in Algorithm 1.

For single-objective problems simulated annealing, proposed by Kirkpatrick et al. (1983), can be regarded as the computational analogue of slowly cooling a metal so that it adopts a minimum-energy crystalline state. In order to minimise a computational energy $E(\omega)$ a computational temperature T is progressively lowered during the optimisation according to an annealing schedule. At high temperature the state changes freely, whereas as the temperature is lowered the state is increasingly confined due to the high energy cost of rearrangement. At each T the SA algorithm aims to draw samples from the equilibrium distribution $\pi_T(\omega) \propto \exp\{-E(\omega)/T\}$. As $T \to 0$ the probability mass of π_T is increasingly concentrated in the region of the global minimum of E, so eventually any sample from π_T almost surely lies at the minimum of E.

The temperature is fixed at T_k during each of the $k = 1, \ldots, K$ epochs; during each epoch L_k samples are drawn from the equilibrium distribution $\pi_T(\omega)$. This is achieved by Metropolis-Hastings sampling

(Metropolis et al., 1953), which involves making proposals ω' (line 4 of Algorithm 1) that are accepted with probability

$$A = \min\left(1, \exp\{-\delta E(\omega, \omega')/T\}\right) \tag{3}$$

where the energy difference between the states ω and ω' is

$$\delta E(\omega, \omega') \equiv E(\omega') - E(\omega). \tag{4}$$

(Lines 6-8 of Algorithm 1.) Initially, when T is high, perturbations from ω to ω' which increase the energy are likely to be accepted (in addition to perturbations which decrease the energy, which are always accepted). This allows the annealer to explore the search space, so as to not become trapped in local minima. As T is reduced only perturbations leading to small increases in E are accepted, so that only limited exploration is possible as the system settles on the global minimum. Convergence to the global minimum of single-objective problems is guaranteed if the cooling schedule is sufficiently gradual (Geman and Geman, 1984) and, although this is infeasibly slow, experience has shown SA to be a very effective optimisation technique even with relatively rapid cooling schedules (Ingber, 1993; Salamon et al., 2002).

For multi-objective problems the usual single-objective algorithm is augmented by the addition of an archive F that is the best estimate of the Pareto front found thus far; that is, it is the non-dominating set whose members are not dominated by any solution visited during the optimisation. Lines 9-12 of Algorithm 1 maintain this archive, which is initialised from the initial state (line 1). Each element of the state that is not dominated by an existing element of the archive is added to the archive (line 12) and any members of F that are dominated by the new entrant are deleted from F (line 11).

2.1 States and perturbations

In many algorithms the state consists of a single solution: $\omega = \{\mathbf{x}\}$. Perturbations are made to the single solution: $\mathbf{x} \mapsto \mathbf{x}'$; and the energy change $\delta E(\omega, \omega') = \delta E(\mathbf{x}, \mathbf{x}')$ is just the difference in energy between the current solution and the perturbation to it.

An alternative is to regard a set of non-dominating solutions as the state. In this case perturbations are made by selecting, say, μ solutions $\mathbf{x}_i$, $i = 1, \ldots, \mu$, from ω, perturbing each one in turn, $\mathbf{x}_i \mapsto \mathbf{x}'_i$, and forming the perturbed state from ω and the perturbed solutions.

A superficially attractive method of deriving the new state would be to form the union of ω and the perturbations, removing those elements of ω' that are dominated by other elements:

$$\omega' = \text{nondom}\left(\omega \bigcup_{i=1}^{\mu} \{\mathbf{x}'_i\}\right). \tag{5}$$

This scheme ensures that ω' is a mutually non-dominating set. However, note that it is unsuitable for a new state because, by construction, no element of ω' can be dominated by an element of ω, which implies that ω' is never inferior to ω so that no exploratory perturbations can be made. In fact, this scheme leads directly to elitist algorithms, as exemplified by Fieldsend and Singh (2002).

Exploratory perturbations to the state are permitted by retaining all the mutually non-dominating perturbations, but removing any elements of ω that either dominate or are dominated by the perturbations:

$$X' = \text{nondom}\left(\bigcup_{i=1}^{\mu} \{\mathbf{x}'_i\}\right) \tag{6}$$

$$\omega' = X' \bigcup \{\mathbf{x} \in \omega \,:\, (\mathbf{x} \not\prec \mathbf{x}' \wedge \mathbf{x}' \not\prec \mathbf{x}) \forall \mathbf{x}' \in X'\}. \tag{7}$$

As illustrated in Figure 1, this scheme ensures that ω' is itself a mutually non-dominating set as well as allowing inferior states to be visited by the search. The efficacy of set-based annealers based on this scheme is investigated in section 3.3.

The energy change, on the basis of which the perturbed set is accepted, is then the difference in energies between the two sets. Methods of calculating the energy change for particular algorithms are discussed below. It should also be noted that the performance of an optimiser can be affected by the manner in which solutions to be perturbed are selected from ω; particular methods are discussed subsequently.

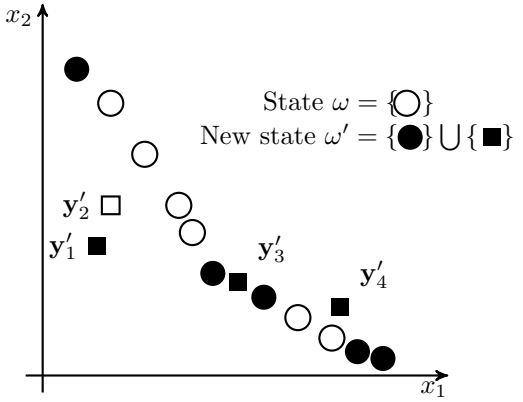

Figure 1. Construction of perturbed state depicted in *objective space*. Circles mark the image under $\mathbf{f}$ of the original state ω and squares mark the image of perturbations to it, $\mathbf{y}'_i = \mathbf{f}(\mathbf{x}'_i)$; elements of ω that are dominated by perturbations or dominate perturbations are indicated by unfilled circles. The new state comprises solutions corresponding to filled circles and filled squares. Note that $\mathbf{y}'_1 \prec \mathbf{y}'_2$.

For multi-objective problems, the set-based state is appealing because the Pareto front itself is generally a set. Since perturbations are made to solutions across the entire estimated front, one may expect the search to proceed more efficiently as it will be less likely to become 'stuck' at a single solution. On the other hand, it might be expected that a single solution state will be efficient because it can rapidly reach the vicinity of the Pareto front (without the need to perturb all the elements of a state to the Pareto front), after which the front is 'filled out' by perturbations transverse to the front; indeed one multi-objective simulated annealing scheme has been observed to work in this fashion (Smith et al., 2008). Finally, it appears that theoretical results for single-objective SA might be adapted to the multi-objective situation using a set-based formulation.

2.2 Greedy Search

Perhaps the simplest search method is greedy search. Greedy methods are defined as those which always make the locally optimal choice: that is, ω' is always accepted if it has a lower energy than ω; perturbations with higher energies are never accepted. Simulated annealing with $T = 0$ may be recognised as a greedy search because there is zero probability that a state with $\delta E(\omega, \omega') > 0$ will be accepted.

Various greedy-type evolutionary algorithms have been developed for multi-objective optimisation, perhaps most prominently those based on $(\mu+\lambda)$-evolution strategies (Knowles and Corne, 1999; Laumanns et al., 2002; Fieldsend et al., 2003; Everson and Fieldsend, 2006). These incorporate algorithms in which the state is a single solution, for example the well-known PAES algorithm (Knowles and Corne, 1999), and those in which the state is comprised of the non-dominated archive from which a single solution is selected at each iteration (Fieldsend et al., 2003). In either case the acceptance criterion is simple: if the perturbation is better than or non-dominated by archive members the candidate is accepted, if it is worse it is rejected.

The strictness of the greedy search acceptance criterion means that the perturbation operator(s) used can greatly influence the results. Since the algorithm can never make moves to regions of higher energy, it can become stuck at a local optimum, relying on a single perturbation to carry the state to a region of lower energy. Such perturbations may be rare (or indeed non-existent), particularly when the dimension of the search space is large, meaning that the global optimum is not found in finite time. The attraction of methods such as simulated annealing is that they permit exploratory perturbations to high energy regions that may subsequently lead to low energy states.

2.3 Energy Functions

The choice of an energy function to compare the quality of two states is fundamental to the operation of simulated annealing and related algorithms. Although in single-objective problems the energy is

determined by the objective to be minimised, there is considerable choice in the multi-objective case. Three major groups of energy function may be identified as follows.

Weighted sum The most obvious method of applying a single-objective optimiser such as simulated annealing to a multi-objective problem is optimise a weighted sum of the objectives. For simulated annealing, using weights w_i, an energy is formed as:

$$E(\mathbf{x}) = \sum_{i=1}^{D} w_i f_i(\mathbf{x}). \tag{8}$$

Several works have investigated this approach using a single solution state: (Serafini, 1994; Ulungu et al., 1999; Czyżak and Jaszkiewicz, 1998; Nam and Park, 2000; Hapke et al., 2000; Tuyttens et al., 2003). Suppapitnarm et al. (2000) employ a weighted product approach is which is equivalent to a weighted sum of the logarithms of the objectives. Generally these procedures always accept a perturbation if it is not dominated by the current archive, but use the composite energy (8) to calculate the acceptance probability if the perturbation is dominated by an element of F.
While convergence to a single point on the Pareto front is guaranteed (as a consequence of Geman & Geman's proof for a scalar SA and provided that the annealing schedule is sufficiently gradual) there are significant drawbacks to this approach. It is not clear how the relative weights of the objectives should be determined in advance, and these will determine which point on the Pareto front is eventually located. A major advantage of multi-objective optimisation is that decisions about the relative importance of the objectives can be made once the algorithm has been executed, subsequent to the discovery of the shape of the front, which is generally not known in advance. In addition, it has been shown that parts of the Pareto front are inaccessible when fixed weights are employed (Das and Dennis, 1997). Recognising this, investigators have proposed a variety of schemes for adapting the w_i during the annealing process to encourage exploration along the front (e.g., Jaszkiewicz, 2001).

Dominance Since the dominance relation is used for comparison of individual solutions in multi-objective problems and is widely used in multi-objective evolution and genetic algorithms, it is desirable to use dominance to compare the quality of solutions. A further property recommending its use is that dominance does not use metric information in the objective space, rendering a dominance-based energy function invariant to rescalings of the objectives.
If the Pareto front $\mathcal{P}$ were available a simple energy function of a state ω could be defined as follows. Let $\mathcal{P}_{\mathbf{x}}$ be the portion of $\mathcal{P}$ that dominates the image of $\mathbf{x}$ under $\mathbf{f}$:

$$\mathcal{P}_{\mathbf{x}} = \{\mathbf{y} \in \mathcal{P} : \mathbf{y} \prec \mathbf{f}(\mathbf{x})\}. \tag{9}$$

Then we might define $E(\omega) = \sum_{\mathbf{x}\in\omega} \mu(\mathcal{P}_{\mathbf{x}})$ where μ is a measure on $\mathcal{P}$. We are principally interested in finite sets approximating $\mathcal{P}$ and so we shall take $\mu(\mathcal{P}_{\mathbf{x}})$ to simply be the cardinality of $\mathcal{P}_{\mathbf{x}}$. This energy has the desired properties. If $\omega \subseteq \mathcal{P}$ then $E(\omega) = 0$ and solutions more distant from the front are, in general, dominated by a greater proportion of $\mathcal{P}$ and so have a higher energy.
This energy does not involve any *a priori* weighting of the objectives. Furthermore, Geman and Geman's (1984) work guarantees convergence to the Pareto front. Unfortunately, the true Pareto front is not available during the optimisation and the difference in energy between two states must therefore be approximated. Dominance-based energy functions utilising only the relative dominance of a solution compared to previously located solutions have been investigated, and shown to significantly out-perform a simulated annealer utilising a weighted sum energy and to compare very favourably with the NSGA-II multi-objective genetic algorithm (Smith et al., 2008). Specific dominance-based energy functions for single solution and set states are described in section 3 and empirically compared in section 4.

Volume It has been proposed (Fleischer, 2003; Zitzler and Künzli, 2004) that a multi-objective simulated annealer can be trivially constructed by defining the energy of a state to be the negative of the volume of objective space dominated by the state.[1] It is envisaged that the state is a set of non-dominated solutions and the volume dominated by this set is clearly maximised when the set equals

[1]Practically, the volume of some hyper-rectangle in objective space dominated by the state must be used to avoid infinite volumes.

Table 1. Multi-objective optimisation algorithms based on simulated annealing. Symbols indicate example published algorithms: † Smith et al. (2008); ‡ Knowles and Corne (1999); ∗ Fieldsend and Singh (2002).

	Dominance energy		Volume energy
	Single $\mathbf{x}$ state	Set state	
Exploratory $T > 0$	MOSA†	SAMOSA	VOLMOSA
Greedy $T = 0$	MOSA0 PAES‡	SAMOSA0 Set-sampled (1+1)-ES∗	VOLMOSA0

the Pareto front. While it is possible to evaluate the volume dominated by a state comprising a single solution, this is equivalent to optimising an unweighted product composite objective function.
While the dominated volume is attractive from a purely theoretical viewpoint, its practical computation turns out to be problematic for more than two objectives. Although not in the context of simulated annealing, Emmerich et al. (2005) have used the dominated volume in a two-objective optimisation algorithm, which has been extended to a three-objectives by Naujoks et al. (2005); Wagner et al. (2007). Recently Bradstreet et al. (2006, 2007, 2008) have explored dominated volume approaches for many objectives using a fast incremental algorithm for hypervolume calculation, and Bader and Zitzler (2008) have used Monte Carlo sampling to derive a solution ranking based on the dominated volume. We examine the performance of the dominated volume in a SA framework for two objective problems in section 4.

3 Algorithms

As described in the previous section, different multi-objective optimisation algorithms related to simulated annealing may be obtained through different choices of state (single solution or set of solutions), energy (weighted sum, dominance-based or volume-based) and whether the search is exploratory (computational temperature $T > 0$) or greedy ($T = 0$). Table 1 summarises greedy and exploratory algorithms using dominance and volume energies, together with single solution and set states, which are described in this section; their performance on standard test problems is compared in section 4. Greedy multi-objective optimisers have appeared in the literature using single solution states (e.g., PAES Knowles and Corne, 1999), and set states (e.g., Fieldsend and Singh, 2002); here direct comparison with exploratory optimisers is facilitated by setting to zero the computational temperature of the exploratory algorithms MOSA (single solution state, dominance energy), SAMOSA (set state, dominance energy) and VOLMOSA (set state, dominated volume energy).

3.1 MOSA: single solution state

The multi-objective simulated annealing algorithm used here was first proposed by Smith et al. (2004, 2008).[2] It maintains a single current solution $\omega \equiv \{\mathbf{x}\}$ which is perturbed to create a new state $\mathbf{x}'$, acceptance of which is determined by the difference in energy between the solutions $\mathbf{x}$ and $\mathbf{x}'$, based upon the proportion of an archive F of previously located non-dominated solutions that dominate $\mathbf{x}$ and $\mathbf{x}'$. The energy change is defined as:

$$\delta E(\omega, \omega') = \frac{1}{|\tilde{F}|}\left(|\tilde{F}_{\mathbf{x}'}| - |\tilde{F}_{\mathbf{x}}|\right) \tag{10}$$

where $\tilde{F} = F \cup \{\mathbf{x}\} \cup \{\mathbf{x}'\}$ is the archive augmented by the state and the perturbation, and $|\tilde{F}_{\mathbf{x}}|$ and $|\tilde{F}_{\mathbf{x}'}|$ are the number of solutions in $\tilde{F}$ dominating $\mathbf{x}$ and $\mathbf{x}'$ respectively. If $\tilde{F}$ is a non-dominating set the energy difference between any two of its elements is zero. Note also that $\delta E(\omega', \omega) = -\delta E(\omega, \omega')$ so that δE may be regarded as an energy difference; furthermore the principle of detailed balance, upon which

[2] We note that Knowles (2002) also suggested a multi-objective simulated annealing procedure in which a single solution is compared with an archive.

Algorithm 2 SAMOSA

Perturbation

$\mathbf{x} := \texttt{Uniselect}(\omega)$
$\mathbf{x}' := \texttt{perturb}(\mathbf{x})$
$\omega' := \{\mathbf{x}'\}$
`foreach` $\mathbf{u} \in \omega$
  `if` $(\mathbf{x}' \not\prec \mathbf{u}) \ \wedge \ (\mathbf{u} \not\prec \mathbf{x}')$
    $\omega' := \omega' \cup \{\mathbf{u}\}$

Energy change

$\delta E := 0$
`foreach` $\mathbf{u} \in \omega$
  `if` $\mathbf{x}' \prec \mathbf{u}$
    $\delta E := \delta E + 1$
  `else if` $\mathbf{u} \prec \mathbf{x}'$
    $\delta E := \delta E - 1$
$\delta E(\omega, \omega') := \delta E / |\omega|$

Metropolis-Hastings sampling relies for sampling from $p_T(\omega)$ is violated if this condition does not hold (Schuur, 1997). The inclusion of the current solution and the proposal in $\tilde{F}$ means that $\delta E(\omega, \omega') < 0$ if $\mathbf{x}' \prec \mathbf{x}$, which ensures that perturbations that move the estimated front towards the true front are always accepted. Proposals that are dominated by one or more members of the current archive are accepted with a probability depending upon the difference in the number of solutions in the archive that dominate $\mathbf{x}'$ and $\mathbf{x}$. In practise when there are few elements in $\tilde{F}$, it is augmented by interpolating from the attainment surface in order to increase the energy resolution (Smith et al., 2008).
Although sophisticated perturbation schemes that adjust the perturbation scale during optimisation may be used with MOSA, a straightforward method, applicable to SAMOSA and VOLMOSA, is employed here. The perturbation scheme is a single-point method in which one of the P decision variables of $\mathbf{x}$ is selected uniformly at random and perturbed by the addition of a random variable ϵ drawn from a heavy-tailed Laplacian distribution, $p(\epsilon) \propto e^{-|\epsilon/\sigma|}$, where the scale factor σ sets the magnitude of the perturbation. In this work $\sigma = 0.1$, corresponding to 1/10th of the range of the decision variables; experiments suggest that the performance of these algorithms is not strongly dependent on the magnitude of σ.

3.2 SAMOSA: set state

The single solution state of the MOSA algorithm typically locates the Pareto front (or a local front) and then moves across it, discovering the extent of the front as it travels. In contrast the SAMOSA algorithm presented here aims to converge a set of solutions covering the front simultaneously.
The state, ω, of the SAMOSA algorithm is a set of mutually non-dominating solutions. This state is not of a fixed size and, since during simulated annealing the set may move away from as well as towards the Pareto front, it is possible that elements of ω may be dominated by previously discovered solutions. The state is initialised with a single, randomly generated solution.
SAMOSA perturbs the state by perturbing a single member of the state, $\mathbf{x} \mapsto \mathbf{x}'$, using the one-point scheme described previously for MOSA. To ensure that ω' is a mutually non-dominating set and that exploratory movements (those that result in the perturbed solution being dominated by element(s) of ω) are possible, the perturbed state is formed by adding to $\mathbf{x}'$ those elements of ω that neither dominate $\mathbf{x}'$ nor are dominated by $\mathbf{x}'$:

$$\omega' = \{\mathbf{x}'\} \bigcup \{\mathbf{x} \in \omega \ : \ \mathbf{x} \not\prec \mathbf{x}' \wedge \mathbf{x}' \not\prec \mathbf{x}\}. \tag{11}$$

It is important that the member of ω which is perturbed is not itself removed from ω unless it is either dominated by or dominates $\mathbf{x}'$, because perturbing the selected member, instead of a copy, would fix the size of the state set at its initial size, one.
Creation of the perturbed state is straightforwardly accomplished as shown in Algorithm 2. The particular member of the ω to perturb is selected using a simple scheme dubbed 'Uniselect'. In the same spirit as the gridding scheme used by PAES (Knowles and Corne, 1999), the aim of Uniselect is to prevent clustering of solutions in a particular region of ω biasing the search because they are selected more frequently. To achieve this, each time a solution is to be selected from ω an objective, i, is chosen at random (with equal probability), a random number u is drawn uniformly in the range $[\min_{\mathbf{x}\in\omega}(f_i(\mathbf{x})), \max_{\mathbf{x}\in\omega}(f_i(\mathbf{x}))]$, and the element of ω to be perturbed is the element with i^{th} coordinate closest to u: $\mathbf{x} = \arg\min_{\mathbf{x}\in\omega} |f_i(\mathbf{x}) - u|$. Although Uniselect has been used here, similar methods (e.g., PQRS, Fieldsend et al., 2003) that counteract the effects of clustering yield comparable results.

The magnitude of the energy difference between ω and ω' is defined as the proportion of the state set ω that is removed in ω':

$$\delta E(\omega, \omega') = \frac{1}{|\omega|} \left[|\{\mathbf{u} \in \omega \,:\, \mathbf{u} \prec \mathbf{x}'\}| \;-\; |\{\mathbf{u} \in \omega \,:\, \mathbf{x}' \prec \mathbf{u}\}| \right]. \tag{12}$$

If solutions are removed from ω due to the insertion of a dominating solution, a negative value is assigned, whereas if solutions are removed due to the insertion of a dominated solution, the assigned value is positive. Note that one or other of the two terms on the r.h.s. of (12) is always zero; if the perturbation does not dominate and is not dominated by any member of ω, then $\delta E(\omega, \omega') = 0$ so that this non-dominating perturbation is accepted. When members of ω' dominate a large proportion of ω, this is considered to be a large improvement, and a large negative $\delta E(\omega, \omega')$ is assigned. Similarly, when ω' is dominated by much of ω, the new state is significantly worse and so is assigned a large positive $\delta E(\omega, \omega')$. The magnitude of the energy change is correctly assigned to be small when there are small differences between ω and ω', and large when one set dominates much of the other. As shown in Algorithm 2, the energy difference is simply calculated by comparing the perturbation with each of the elements in ω and may be combined with the formation of ω'.
Note that if F is a subset of the Pareto front $F \subseteq \mathcal{P}$ then $\delta E(F, \omega) \geq 0$, with equality if and only if all elements of ω are not dominated by $\mathcal{P}$, so that the global optimum has minimum energy among all feasible sets. However, we note that in general $\delta E(\omega, \omega') \neq -\delta E(\omega', \omega)$ and δE does not satisfy the triangle inequality so there is little hope of proving convergence of an algorithm based upon it.

3.3 VOLMOSA: set state, dominated volume energy

Although, as noted above, Fleischer (2003) suggested basing a multi-objective simulated annealer on the dominated volume, no report on the performance of such an annealer has appeared in the literature. Nevertheless, an annealer with a set state is straightforwardly defined. Here the state, initialisation and formation of the perturbed state are identical to those of SAMOSA; the only difference in the algorithms is in the assessment of the energy difference. The energy $E(\omega)$ of a state ω is defined to be the volume of the hyper-rectangle defined by the origin and some reference point $\mathbf{r} \in \mathbb{R}^D$ that is dominated by all elements of ω. The energy difference is then just the difference in energies of the states: $\delta E(\omega, \omega') = E(\omega) - E(\omega')$. Note that since it is only the energy *difference* that is of interest, $\mathbf{r}$ can be moved during annealing so long as it is dominated by all the elements of ω and ω'.
A volume-based annealer is attractive because the energy itself $E(\omega)$ is defined, with the minimum energy corresponding to the Pareto set. This scalarization means that all the existing theory for single-objective SA is directly applicable.
The time complexity given by Huband et al. (2005) for Fleischer's algorithm (2003) to exactly calculate the dominated volume for ω is $\mathcal{O}(|\omega|^D)$. Faster algorithms with complexities of $\mathcal{O}(|\omega|^{D-1})$ (Huband et al., 2005), $\mathcal{O}(|\omega|^{D-2} \log |\omega|)$ (Fonseca et al., 2006) and $\mathcal{O}(|\omega| \log |\omega| + |\omega|^{D/2})$ (best worst case performance (Beume and Rudolph, 2006)) are still expensive for incorporation in algorithms which must repeatedly evaluate the volume. Monte Carlo sampling, an approximate method, has been used for comparing fronts (Fieldsend et al., 2003), but when the difference in volume between ω and ω' is small it is again prohibitively expensive to achieve the necessary accuracy. Note however that Bader and Zitzler (2008) have used Monte Carlo sampling to give an approximate *ranking* of solutions based on the dominated volume.
Although the cost of calculating the volume dominated by a complete set ω is high, in evolutionary algorithms, and VOLMOSA in particular, the set of interest ω' commonly differs from ω only by a single solution. Taking advantage of this, here an incremental method for dominated volume calculation was used (Smith, 2006); see also the recent work of Bradstreet et al. (2007). The method used here is an extension of Fleischer's (2003) calculation, presented as a recursive method, instead of explicitly stack-based, and extended such that the set does not need to be mutually non-dominating. Using this method, when a perturbation to ω generates a solution which is not dominated by the members of ω (that is, when the volume is not decreased), only the difference in volume between the two sets need be calculated, which is considerably simpler than the evaluation of the volume dominated by the entire set.

4 Comparisons

In this section the performance of MOSA and SAMOSA are compared with each other and with their greedy ($T = 0$) versions. Results on MOSA and SAMOSA give a direct comparison of single solution states against set states, while dominance-based and volume-based energy measures are compared via the SAMOSA and VOLMOSA algorithms. As displayed in Table 1, the temperature zero versions of the algorithms are denoted by MOSA0 and SAMOSA0.

Performance is evaluated on well-known test functions from the literature, namely the DTLZ test suite problems 1-6 (Deb et al., 2001, 2002b)[3]. The number of decision variables used are those recommended by Deb et al. (2002b), namely: DTLZ1 7; DTLZ2 7; DTLZ3 12; DTLZ4 12; DTLZ5 12; DTLZ6 22. The benefit of using these test functions is that the true Pareto front $\mathcal{P}$ is known, so the proximity of the estimated Pareto front, F, can be measured. Results with $D = 3$ objectives are shown for the MOSA and SAMOSA comparisons, but computational expense limits SAMOSA–VOLMOSA comparisons to two objective problems.

All annealers used a common annealing schedule. All epochs were of equal length, $L_k = 100$ and the temperature was reduced according to $T_k = \beta^k T_0$, where β was chosen so that T_k is 10^{-5} after approximately two thirds of the function evaluations are completed. 50000 function evaluations were used for each of the problems, except the DTLZ2 problem for which 5000 evaluations were used due to the ease of convergence on this problem. The MOSA initial temperature was set so that approximately half of perturbations are accepted, as described by Smith et al. (2008), while for SAMOSA $T_0 = 4$ which results in an initial acceptance rate of roughly 0.5.

Twenty runs from different (random) initialisations were made for each algorithm. In addition to statistical summaries, we present representative results on each problem. The median (to reduce the influence of outliers) distance of solutions in F to $\mathcal{P}$ from a single run is calculated:

$$\bar{d}(F, \mathcal{P}) = \underset{\mathbf{x} \in F}{\text{median}}(d(\mathbf{x}, \mathcal{P})) \tag{13}$$

where $d(\mathbf{x}, \mathcal{P})$ is the minimum Euclidean distance between $\mathbf{x}$ and the true front $\mathcal{P}$. The representative run is then chosen as the run with the median $\bar{d}(F, \mathcal{P})$ over the 20 runs; denoting the front resulting from the ith run as F_i, the representative run is:

$$\bar{F} = \underset{F_i \text{ for } i=1,\ldots,20}{\arg\text{median}} \; (\bar{d}(F_i, \mathcal{P})) \tag{14}$$

and the overall median distance is reported as

$$\bar{d} = \underset{F_i \text{ for } i=1,\ldots,20}{\text{median}} (\bar{d}(F_i, \mathcal{P})). \tag{15}$$

This measure depends on the relative scalings of the objective functions, but here it gives a fair comparison because the objectives for the DTLZ functions all have similar ranges. We also remind the reader that the distance function does not completely summarize the quality of an approximation to the Pareto front because, for example, a few distant points may contribute strongly to the measure. Recognising this, we also present performance summaries using the dominated volume measure and the unary ϵ-indicator. The dominated volume measure $\mathcal{V}(\mathcal{P}, F)$ quantifies the amount of objective space that is covered by the true front, but not by the archive; it thus measures the coverage of $\mathcal{P}$ by F rather than merely the proximity of F to $\mathcal{P}$. When F covers the entire Pareto front then $\mathcal{V}(\mathcal{P}, F) \approx 0$, but an archive comprised of a few solutions clustered together on the true front will have a larger $\mathcal{V}(\mathcal{P}, F)$ than an archive of solutions well spread across the front and therefore dominating a larger fraction of objective space. To make connection with other work the volume measure is denoted by $\mathcal{V}(\mathcal{P}, F)$, but it may be recognised as being proportional to the energy $E(\mathcal{P}, F)$ used in the VOLMOSA algorithm. It was calculated by Monte Carlo sampling of 10^5 samples in the hyper-rectangle defined by the origin and the extremal elements of $\mathcal{P}$. The (additive) unary ϵ-indicator measure $I_\epsilon(F)$ quantifies the minimum amount by which $\mathcal{P}$ must be translated in each of the objective coordinates in order to that all of the elements of $\mathcal{P}$ are dominated by an element of F (Zitzler et al., 2003).

Files containing archives located by the algorithms are available at `http://www.secam.ex.ac.uk/people/staff/reverson/research/mosa.html`.

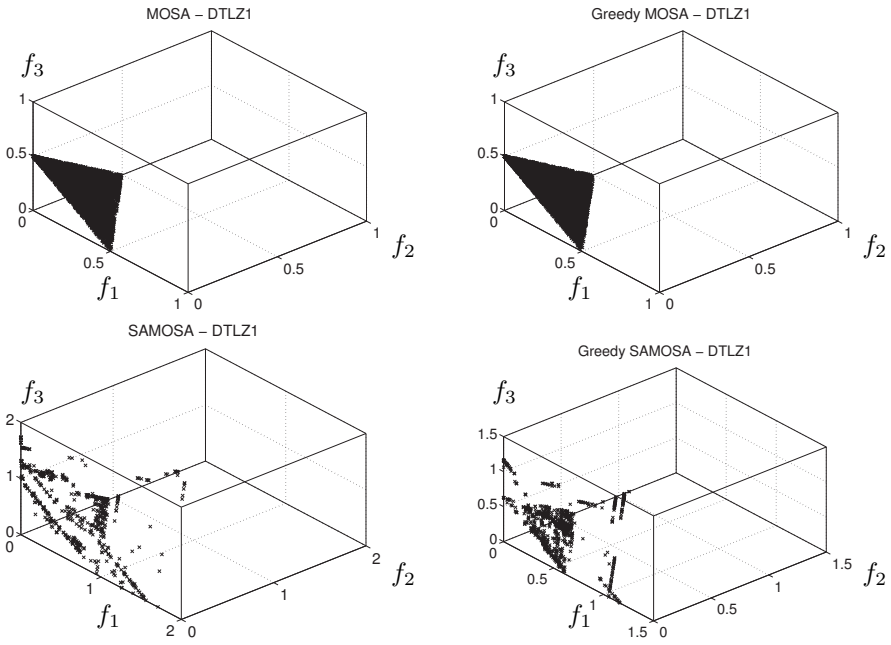

Figure 2. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ1 after 50000 function evaluations.

DTLZ 1 Figure 2 presents results for both the greedy and non-greedy versions of both the MOSA and SAMOSA algorithms on the DTLZ1 problem. The true front for the DTLZ1 problem is the plane intersecting the axes at $(0, 0, 0.5)$, $(0, 0.5, 0)$ and $(0.5, 0, 0)$, and with $f_i \geq 0$. Both the MOSA (exploratory) and MOSA0 (greedy) algorithms converge well to the true front and provide good coverage of the front. SAMOSA and SAMOSA0 both perform less well than the MOSA variations, not coming so close to the front and not covering it so uniformly.

Statistical quantification, using the non-parametric Mann-Whitney U-test, of the comparative performance of the algorithms is shown in Table 2. The table divides into three parts, one for each performance metric, $\bar{d}(F, \mathcal{P})$, $\mathcal{V}(\mathcal{P}, F)$ and $I_\epsilon(F)$. Each cell shows the number of times the algorithm in the first column is superior to that in the second column, from a possible $20^2 = 400$ comparisons. Bold (italic) entries indicate that, according to the Mann-Whitney U-test, the null hypothesis that the two algorithms yield the same median performance can be rejected at the 0.005 (0.025) level.

The formulation of DTLZ1 means that it is easy to make perturbations which traverse the front, so there is no gain from SAMOSA's approach of simultaneously optimising a set of solutions by making perturbations to many solutions thus bringing them all to the true front together. MOSA's approach of optimising a single solution, in this case, results in the front being located more quickly, allowing time to traverse the front thus providing the good coverage exhibited. Since the estimated front is still converging, it can be expected that, if given sufficient time, SAMOSA would also converge on the true front but that it will require significantly more evaluations. Particularly interesting is that, despite what are described by Deb et al. (2001, 2002b) as $\approx 10^{11}$ local fronts arranged parallel to $\mathcal{P}$ in the DTLZ1 problem, the greedy algorithms are able to converge to, or near to, the true front; MOSA0 out-performs MOSA and SAMOSA0 out-performs SAMOSA. This demonstrates that local fronts in this form do not necessarily inhibit convergence of greedy algorithms – we discuss this further in section 5.

DTLZ 2 Results for DTLZ2 are presented in Figure 3. Here $\mathcal{P}$ is the positive octant of a sphere of radius 1, centred on the origin. As may be expected on this easy problem, the greedy variants significantly out-perform their respective non-greedy counterparts: on the median run 95% of solutions for MOSA are within 2.5×10^{-3} of $\mathcal{P}$ compared with 1.2×10^{-4} for MOSA0; and closer than 1.1×10^{-2} for SAMOSA compared with 4.1×10^{-3} for SAMOSA0. Again the MOSAs out-perform the SAMOSAs as the ease of

[3]The formulations of the DTLZ functions presented in (Smith et al., 2008) are used because these correct typographical errors in the formulae presented in (Deb et al., 2001) and (Deb et al., 2002b).

Table 2. Pair-wise comparisons of the MOSA, MOSA0, SAMOSA and SAMOSA0 algorithms on the DTLZ problems.

		DTLZ1	**DTLZ2**	**DTLZ3**	**DTLZ4**	**DTLZ5**	**DTLZ6**
Distance $\bar{d}(F,\mathcal{P})$							
MOSA	MOSA0	42	124	138	144	116	159
	SAMOSA	**400**	**400**	**400**	**400**	**400**	**400**
	SAMOSA0	**400**	**400**	**400**	**315**	**400**	**400**
MOSA0	MOSA	**358**	*276*	262	256	*284*	241
	SAMOSA	**400**	**400**	**400**	**380**	**400**	**400**
	SAMOSA0	**400**	**400**	**400**	**309**	**400**	**400**
SAMOSA	MOSA	0	0	0	0	0	0
	MOSA0	0	0	0	20	0	0
	SAMOSA0	58	74	115	3	87	18
SAMOSA0	MOSA	0	0	0	85	0	0
	MOSA0	0	0	0	91	0	0
	SAMOSA	**342**	**326**	*285*	**397**	**313**	**382**
Volume $\mathcal{V}(\mathcal{P},F)$							
MOSA	MOSA0	52	143	88	**400**	211	**307**
	SAMOSA	**400**	**397**	**400**	40	101	**400**
	SAMOSA0	**400**	**334**	**400**	205	69	**400**
MOSA0	MOSA	**348**	257	**312**	0	189	93
	SAMOSA	**400**	**365**	**400**	0	91	**392**
	SAMOSA0	**400**	**327**	**400**	26	66	**371**
SAMOSA	MOSA	0	3	0	**360**	**299**	0
	MOSA0	0	35	0	**400**	**309**	8
	SAMOSA0	58	15	200	**340**	127	39
SAMOSA0	MOSA	0	66	0	195	**331**	0
	MOSA0	0	73	0	**375**	**334**	29
	SAMOSA	**342**	**385**	200	60	*273*	**361**
Unary ϵ**-indicator**							
MOSA	MOSA0	92	175	135	**400**	208	**336**
	SAMOSA	**400**	226	**400**	40	4	**400**
	SAMOSA0	**400**	231	**400**	206	0	**391**
MOSA0	MOSA	**308**	225	*265*	0	192	64
	SAMOSA	**400**	241	**400**	0	11	**344**
	SAMOSA0	**400**	241	**400**	19	1	*270*
SAMOSA	MOSA	0	174	0	**360**	**396**	0
	MOSA0	0	159	0	**400**	**389**	56
	SAMOSA0	50	198	134	**346**	65	30
SAMOSA0	MOSA	0	169	0	194	**400**	9
	MOSA0	0	159	0	**382**	**399**	130
	SAMOSA	**350**	202	*266*	54	**335**	**370**

traversal across the front results in redundant perturbations of state members. Put another way: the set-based annealer's potential advantage of being able to perturb solutions in diverse parts of decision space in order to move the front forward if a single solution is 'stuck' on a local front is of no advantage, and in fact perturbing solutions across the entire front is a waste of computational resources on this problem.

DTLZ 3 The DTLZ3 results, presented in Figure 4, show that even on this problem, with many local fronts on which a greedy optimiser might be expected to become stuck, the greedy algorithms are able to out-perform the corresponding non-greedy variants and converge on the true front, which is again a radius 1 octant of a sphere centred on the origin. Due to the large number of similar perturbations that must be made to move an entire set state SAMOSA is again out-performed by MOSA. Note, however, that the perturbation scheme is local in the sense that only one decision variable is perturbed at a time, but global in the sense that there is a finite probability that a perturbation can move the selected decision variable across its entire feasible range.

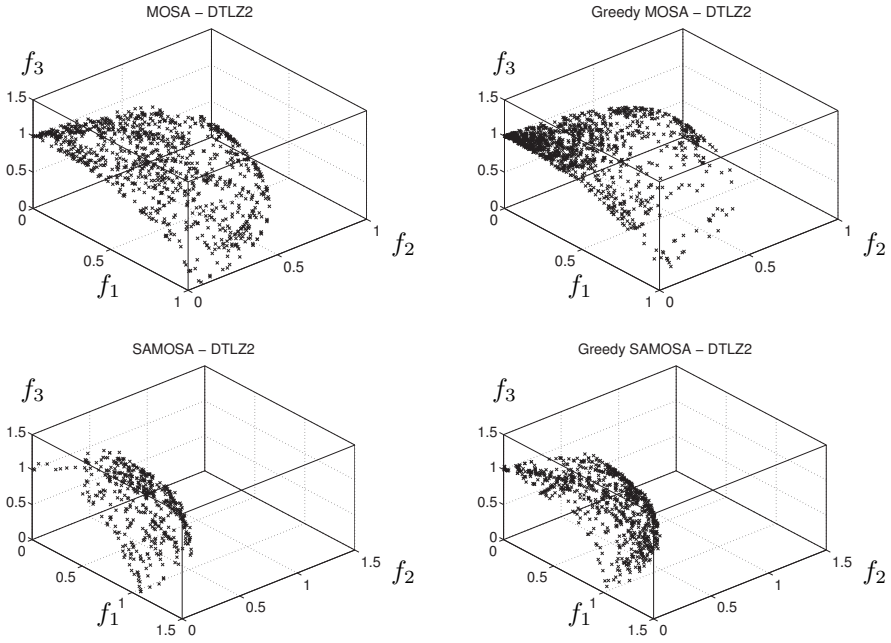

Figure 3. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ2 after 5000 function evaluations.

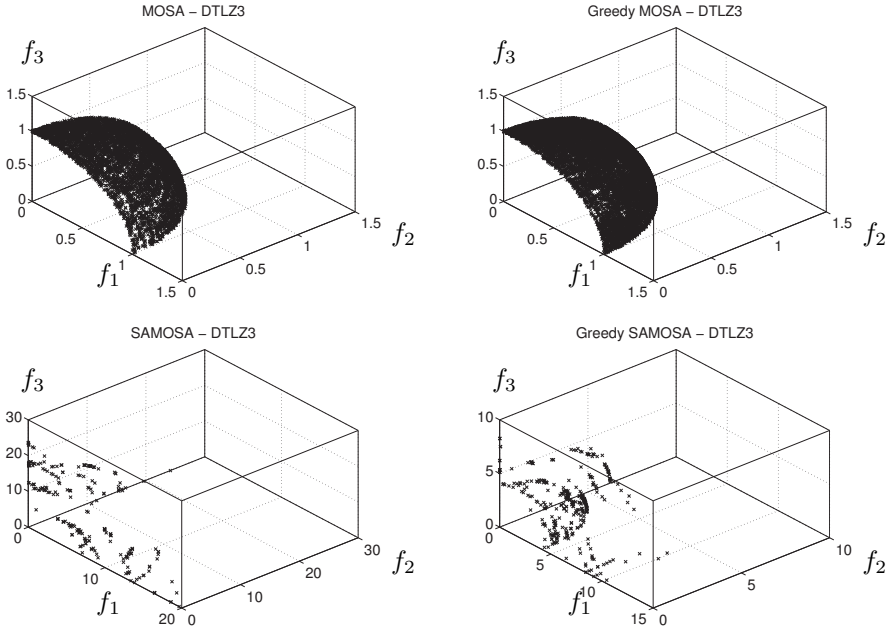

Figure 4. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ3 after 50000 function evaluations.

DTLZ 4 Figure 5 presents the results for DTLZ4. The true front for DTLZ4 is, as for DTLZ2 and DTLZ3, an octant of a unit sphere centred on the origin; the structure of DTLZ4 is such, however, that it is easy for an optimiser to reach one corner of the front, but that the rims correspond to a very small region of decision space and the central region of the front is mapped to yet a smaller region of decision space (Smith et al. (2008) give a more detailed analysis). Here the Uniselect method for selecting solutions for perturbation, combined with the exploratory or non-greedy ability to move away from the front, allows SAMOSA to cover the central region of the front well, a small distance away from $\mathcal{P}$, providing far better coverage of, although less close to, the true front than the other algorithms.

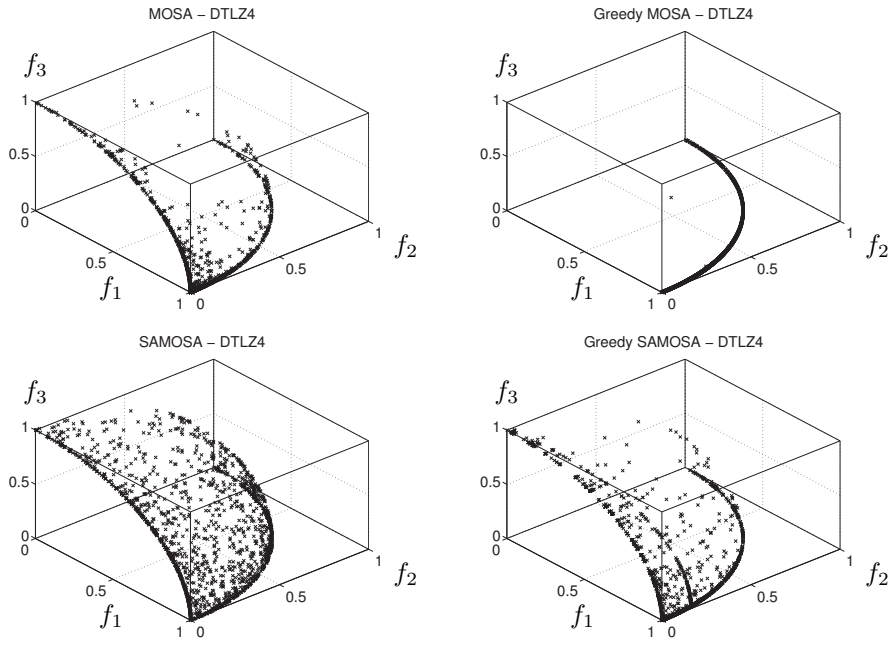

Figure 5. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ4 after 50000 function evaluations.

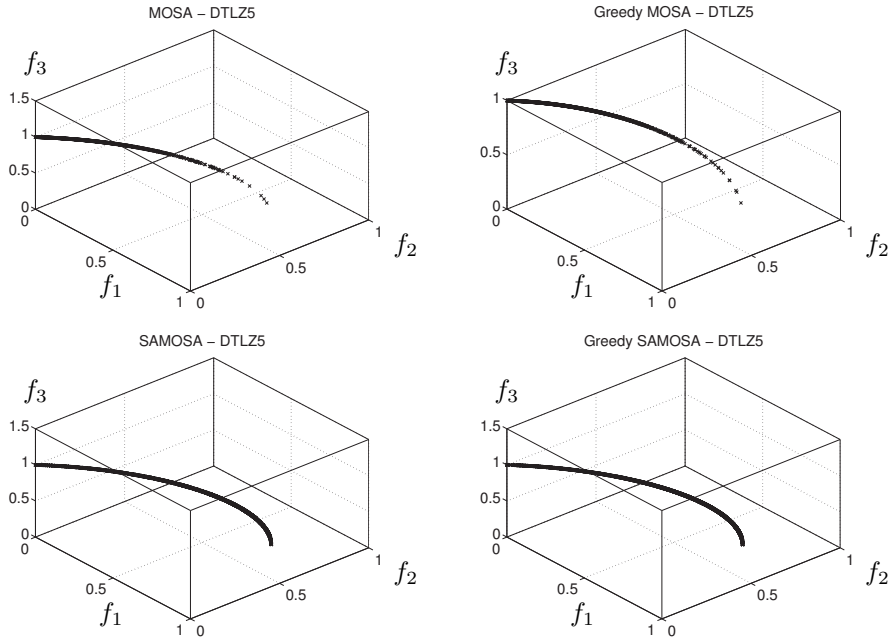

Figure 6. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ5 after 50000 function evaluations.

SAMOSA0 also provides a reasonable coverage of the front, due to the Uniselect selection method. As MOSA uses a single solution for the state, there is no possibility to use a Uniselect method for selecting solutions to perturb evenly across the front, but it has still managed to generate a small covering of the front, possibly through non-greedy moves away from the corner. The MOSA0 variant, having neither Uniselect nor the ability to make non-greedy moves, was unable to leave one of the rims of the front but was still able to converge to within a small distance from the true front for the regions it covered. As shown in Figure 8 and Table 2, this results in different orderings for the algorithms depending upon the assessment metric used: because MOSA has converged very close to a small region of the front it is

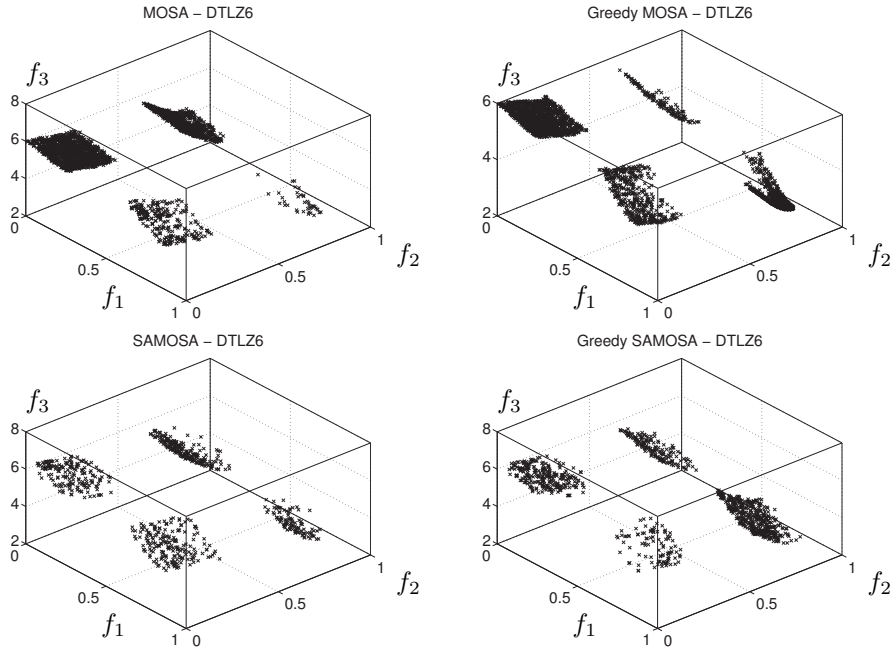

Figure 7. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on DTLZ6 after 50000 function evaluations.

ranked above SAMOSA according to the median distance from the front, whereas the better coverage by SAMOSA leads to it being ranked above MOSA according to the dominated volume and unary ϵ measures.

DTLZ 5 All four algorithms were able to converge well to the true front for DTLZ5, as shown in Figure 6. The true front in this case is a one-dimensional curve, rather than a two-dimensional surface. Uniselect allowed SAMOSA and SAMOSA0 to generate a more even coverage of the front while MOSA and MOSA0 converged closer to the true front. On average the closer convergence of MOSA and MOSA0 leads to them being ranked above SAMOSA and SAMOSA0 according to the median distance measure (see Figure 8 and Table 2), but the ordering is reversed according to the dominated volume and unary ϵ measures, which take into account the more even coverage produced by the set-based algorithms.

DTLZ 6 Figure 7 presents the results for the DTLZ6 problem, the true front for which consists of four disjoint regions or 'cushions' in objective space, to which all four algorithms were able to converge well. Note that MOSA and MOSA0 whose single solution state might be expected to limit them to a single cushion are able to locate all four regions; if the regions in decision space corresponding to each of the objective space cushions were more widely separated then it might be more difficult for the MOSA and MOSA0 annealers to locate all the cushions. The Uniselect selection seems to have enabled SAMOSA and SAMOSA0 to generate a more uniform coverage of the four cushions, while MOSA and MOSA0 converged closer to the true front, and generated more solutions in their archives possibly due to the 'wasteful' effect of making similar perturbations across the range of solutions, as observed on DTLZ2.

4.1 Summary

Statistical summaries are presented in Figure 8 as box plots showing the distributions over 20 runs of the median frontal distance $\bar{d}(F, \mathcal{P})$ from the true Pareto front, the proportion of the volume of the bounding box of $\mathcal{P}$ that is dominated by $\mathcal{P}$ but not by F, and the unary ϵ-indicator $I_\epsilon(F)$.
These results primarily reveal three interesting traits. First, common across these problems, is that the MOSA and MOSA0 algorithms out-perform SAMOSA and SAMOSA0 in their ability to generate solutions very close to the Pareto front. This may be understood as follows: SAMOSA may advance the state by perturbing each of the constituent solutions to move the ω incrementally forward, however,

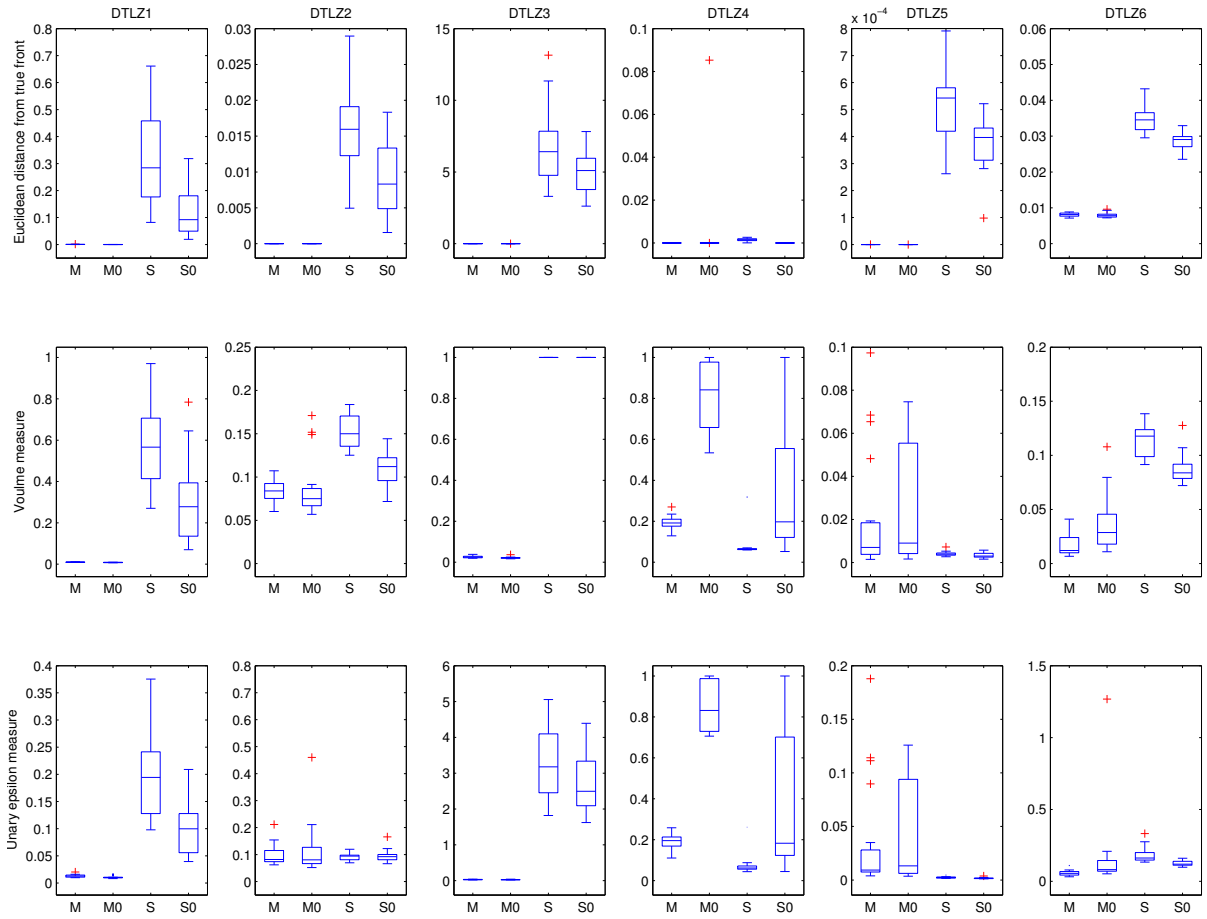

Figure 8. Top: Box plots showing the distribution over 20 runs of the median distances $\bar{d}(F, \mathcal{P})$ of the front from the true Pareto front. 'M', 'M0', 'S' and 'S0' denote the MOSA, MOSA0, SAMOSA and SAMOSA0 algorithms respectively. **Middle:** Box plots showing the distribution over 20 runs of the proportion of the bounding box of the Pareto front dominated by the true front but not the archives. **Bottom:** Box plots showing the distribution over 20 runs of the unary ϵ indicator, $I_\epsilon(F)$ (Zitzler et al., 2003).

if a subsequent perturbation to a solution dominates several other solutions then these solutions are replaced in ω by the dominating solution and the perturbations that have advanced them thus far are wasted; this does not occur for MOSA which focuses on a single solution. It appears from these results that SAMOSA's advantage of being able to perturb solutions in several locations is outweighed by these wasted perturbations.

Second, in contrast to the first trait, the SAMOSA and SAMOSA0 algorithms were able to generate fronts with a more uniform coverage in objective space than MOSA and MOSA0, particularly on the DTLZ4 problem and to a lesser extent on the DTLZ5 and DTLZ6 problems. The Uniselect method, which selects solutions for perturbations from regions of the front uniformly, without prejudice regarding the density of solutions in those regions, maintains a more uniform coverage within the estimated front.

Third, the most surprising comparisons are those between the greedy and non-greedy algorithms. The DTLZ1 and DTLZ3 problems were specifically constructed with the characteristic of many local fronts (Deb et al., 2001, 2002b), designed such that greedy algorithms may become unable to escape them; nonetheless the greedy algorithms are able to out-perform the non-greedy algorithms on these problems. These results make it clear that for many problems, even those with local fronts, greedy algorithms are sufficient; this behaviour is discussed further in section 5 and a problem where this is not true is investigated.

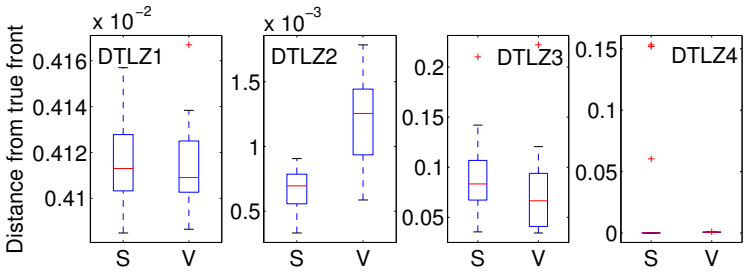

Figure 9. Box plots showing the distribution over 20 runs of the median distances $\bar{d}(F)$ of the front from the true Pareto front. 'S' and 'V' denote the SAMOSA and VOLMOSA algorithms respectively.

4.2 Comparison of Volume and Dominance measures

Comparison of the dominance-based energy measures used in SAMOSA and the volume-based energy measures used in VOLMOSA is prohibitively expensive for even three objectives due to the computational complexity of the dominated volume calculation (Huband et al., 2005; Fonseca et al., 2006), even using an incremental algorithm (Smith, 2006). Comparisons were therefore made using use two-objective versions of the problems[4] and a limited number of evaluations. We draw attention to recent works using the volume measures by Bradstreet et al. (2006, 2007); Wagner et al. (2007), which—unlike this work—use limited size archives. 30000 function evaluations were used for each of DTLZ1, DTLZ3 and DTLZ4 and 5000 evaluations on DTLZ2 due to the ease of convergence on it. The same methodology was used here as for the three-objective comparisons, with 20 runs of each configuration. Again, an annealing schedule was chosen so as to reduce from $T_0 = 4$ initially to $T = 10^{-5}$ after approximately two thirds of the evaluations.

On all these problems the VOLMOSA and SAMOSA energy functions yield very similar results and the annealers converge close to the front. This is confirmed by the box plots shown in Figure 9, which show no essential difference between the performance of the two algorithms. It thus appears that either energy function is equally effective with a set-based multi-objective simulated annealer.

However, as the volume measure uses objective values directly, it may be sensitive to the objective scales in a similar way to composite objective functions; this may change the rate of convergence but not the ability to eventually converge.

5 Greedy Search

The DTLZ1 and DTLZ3 problems present an optimisation algorithm with many local fronts within which it is possible for a greedy-search algorithm to become trapped, yet it is clear from the results presented earlier that the presence of local fronts is not sufficient to prevent a greedy search technique from optimising a problem. Further study with the algorithms presented in this paper reveals a similar situation for the WFG problems (Huband et al., 2005); problems with many local fronts do not necessarily cause greedy algorithms to become 'stuck'. It is not immediately clear, however, why this should be, because the presence of a local front implies that a solution cannot travel through the front (in objective space) by continuing on the same vector (in decision space) that caused it to reach the front. This section discusses the ability of a greedy optimiser to pass through these local fronts and presents a new local-fronted problem on which greedy optimisers perform less well.

Unlike single-objective optimisation problems, where the concepts of local minima and basins are simple, the ideas of local fronts and the associated basins in multi-objective test problems are not necessarily intuitive. In a search space X, a local front is the image under $\mathbf{f}$ of a *local Pareto-optimal set*, $L \subseteq X$, meaning that every neighbourhood of $\mathbf{x} \in L$ is dominated by an element of L (Deb, 1999). For continuous search spaces considered here the neighbourhood of $\mathbf{x}$ is defined as a ball of radius ϵ centred at $\mathbf{x}$ in the limit $\epsilon \to 0$. A greedy search algorithm that takes infinitesimally small steps will be unable to escape from the vicinity of a local front because all perturbations are dominated by L. This is illustrated in Figure 10

[4]The number of decision variables P was chosen as recommended by Deb et al. (2002b), namely: DTLZ1 6; DTLZ2 6; DTLZ3 11; DTLZ4 11.

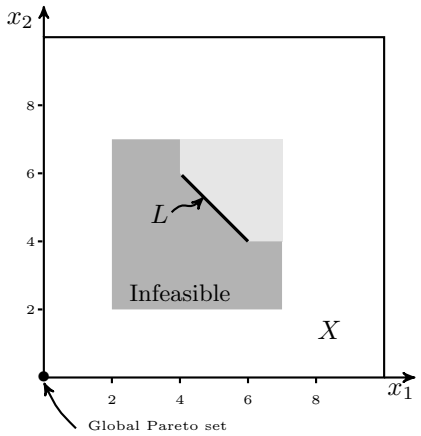

Figure 10. Search space for the problem $\min(x_1, x_2)$ with a local front L. The feasible region is the square $[0, 10] \times [0, 10]$ except for the region marked in dark grey. The global Pareto optimum is the origin and the line marked L is a local Pareto optimum. A greedy searcher taking infinitesimal steps cannot escape the light grey region as it is dominated by L, but a greedy searcher able to take steps of length $\epsilon = 2$ can leave the light grey region and locate the global optimum.

which shows the two-dimensional search space for the problem $\min(x_1, x_2)$ on the feasible region shown.[5] The global Pareto set is the origin, but there is also a local Pareto optimal front L, marked in bold. Once the search has located the ends of L at $(6, 4)$ and $(4, 6)$, infinitesimal perturbations to an $\mathbf{x}$ in the light grey region are dominated by L and a greedy search making infinitesimally small steps cannot escape from the light grey region. On the other hand, a greedy searcher able to make sufficiently large steps can escape the light grey region and arrive at the global Pareto set, either by taking a large vertical or horizontal step and going around the side of the infeasible region or by taking a larger step directly over the infeasible region. We therefore characterise a problem as ϵ *greedy searchable* if from every $\mathbf{x}_0 \in X$ there exists a sequence $\{\mathbf{x}_n\}$ whose limit is the global Pareto set with no $\mathbf{x}_n$ being dominated by its predecessor and such that $\|\mathbf{x}_n - \mathbf{x}_{n-1}\| < \epsilon$ for all n. The problem illustrated in Figure 10 is greedy searchable for $\epsilon > 2$ because all points in the light grey region are no more than a distance 2 from the top or right edges.

Note that set-state greedy searchers may also be able to locate the global Pareto front because, even though elements of the set may become stuck on local fronts, other elements of the state may be able to travel 'around' the local front.

In the extreme case, a greedy-search algorithm allowed unbounded perturbations may make a single perturbation from any solution directly to the Pareto front. While this means that a real greedy optimiser may be able to locate the Pareto front even if the problem is not infinitesimally greedy-searchable, the computational effort required is equivalent to an exhaustive search. MOSA0 and SAMOSA0 are able to locate the Pareto front on the DTLZ problems, despite the presence of local fronts, because the perturbations required to escape these local fronts are relatively small. Although the $T > 0$ versions of MOSA and SAMOSA are able to make non-greedy, exploratory moves, these are of little benefit to them because, with finite perturbations, it is relatively easy to escape from the local fronts without an intermediate, dominated step.

5.1 A Non-infinitesimally Greedy Searchable Problem (NGS)

In order to explore the influence of local fronts in multi-objective problems we present a test problem which is susceptible to easy analysis and which is clearly not infinitesimally greedy-searchable. Moreover it has large regions, analogous to the light grey region in Figure 10, accompanying its local fronts, so that it is very unlikely, or impossible, to make a single perturbation which escapes them.

To visualize the structure of our test problems we first present a reduced version in which there are $D = 2$ objectives and $P = 2$ decision variables, x_1 and x_2, and whose structure is controlled by an

[5] We are grateful to an anonymous reviewer for this problem.

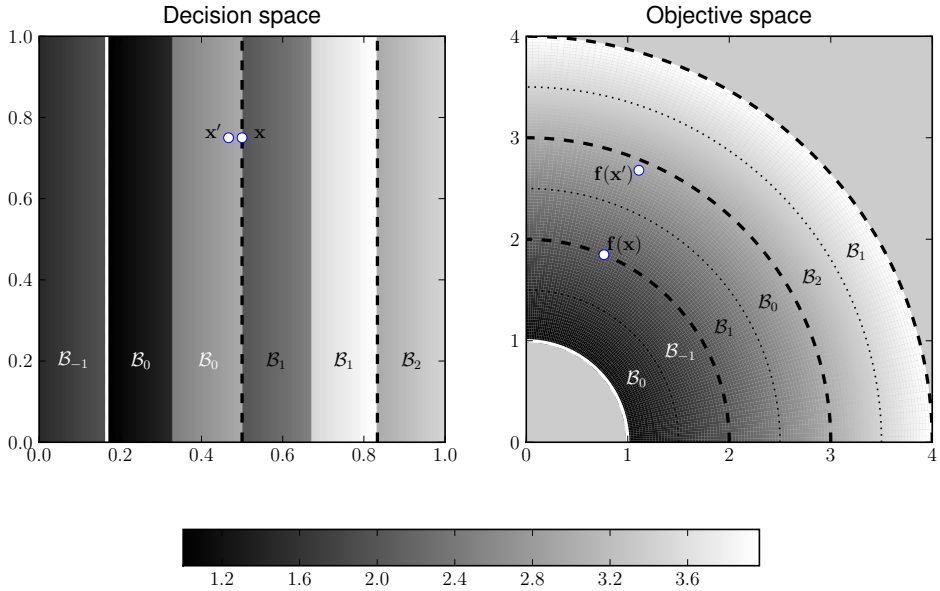

Figure 11. The mapping between decision space and objective space for the reduced problem defined by (16) with $Q = 6$. The grey scale shows the distance of a solution from the origin in objective space. Regions $\mathcal{B}_j$ shown in decision space map to the same coloured regions in objective space. Local fronts at the boundaries of $\mathcal{B}_j$ are marked with dashed lines; the Pareto set at $1/Q \approx 0.166$ in decision space and the corresponding front, the circular arc with $\|\mathbf{f}\| = 1$, are marked with white solid lines. The regions $\|\mathbf{f}\| < 1$ and $\|\mathbf{f}\| > 4$ are infeasible. Small white circles indicate a solution $\mathbf{x} \in L_1$ and a perturbation to it $\mathbf{x}' \in \mathcal{B}_0$; as shown in the objective space panel, $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}')$.

integer parameter Q. The problem is defined by $\mathbf{f}(\mathbf{x}) : [0,1]^2 \mapsto \{\mathbf{y} : 1 \le \|\mathbf{y}\| \le \lceil Q/2 \rceil\}$ where:

$$R(r) = (r - \lfloor r \rfloor + 1 - \text{remainder}(\lfloor r \rfloor, 2))/2 + 1 + \lfloor r/2 \rfloor, \tag{16a}$$
$$f_1(\mathbf{x}) = R(Qx_1)\cos(\pi x_2/2), \tag{16b}$$
$$f_2(\mathbf{x}) = R(Qx_1)\sin(\pi x_2/2). \tag{16c}$$

The mapping defined by this reduced problem is illustrated in Figure 11. As can be seen from the figure, x_1 controls the distance of $\mathbf{f}(\mathbf{x})$ from the origin in objective space, while x_2 determines the ratio of the objectives or equivalently the angular location of $\mathbf{f}(\mathbf{x})$ in objective space. As illustrated by the shading in Figure 11, the mapping between decision space and objective space is discontinuous, which leads to the formation of $\lceil Q/2 \rceil$ distinct regions labelled $\mathcal{B}_j$ in the figure. In decision space $\mathcal{B}_j$ is

$$\mathcal{B}_j = [\max(0, (2j+1)/Q), \min((2j+3)/Q, 1)) \times [0,1] \qquad -1 \le j \le \lfloor Q/2 \rfloor. \tag{17}$$

Note that each region maps to two disconnected regions of objective space, as shown in the figure. Infinitesimal perturbations in decision space from the right hand half of $\mathcal{B}_j$ to the left hand half (from a lighter shade to a darker shade in Figure 11) are possible in a greedy search algorithm because the perturbed solution dominates the unperturbed solution. However, infinitesimal perturbations across the left hand boundary of $\mathcal{B}_j$ (at $x_1 = 2(j+1)/Q$) are not possible and these boundaries are local Pareto sets, L_j. The non-dominated boundary of $\mathcal{B}_0$ is the global Pareto set and the Pareto front is the arc $\|\mathbf{f}\| = 1$.

To see that perturbations across the local Pareto fronts are dominated consider a solution $\mathbf{x}$ lying on the local Pareto front for $\mathcal{B}_j$ so that $x_1 = 2(j+1)/Q$. If $\mathbf{x}$ is perturbed to $\mathbf{x}' \in \mathcal{B}_{j-1}$, then $\mathbf{x}'$ is dominated by $\mathbf{x}$ provided that the perturbation is sufficiently small as illustrated for $j = 1$ in Figure 11. In fact, the solution $\mathbf{x} = ((2j+1)/Q, x_2)$ dominates at least the region

$$\left[\frac{2j-1}{Q}, \frac{2j}{Q}\right) \times \left[\frac{1}{\pi}\arcsin\left(\frac{2j}{2j+1}\sin\frac{2x_2}{\pi}\right), \frac{1}{\pi}\arccos\left(\frac{2j}{2j+1}\cos\frac{2x_2}{\pi}\right)\right] \tag{18}$$

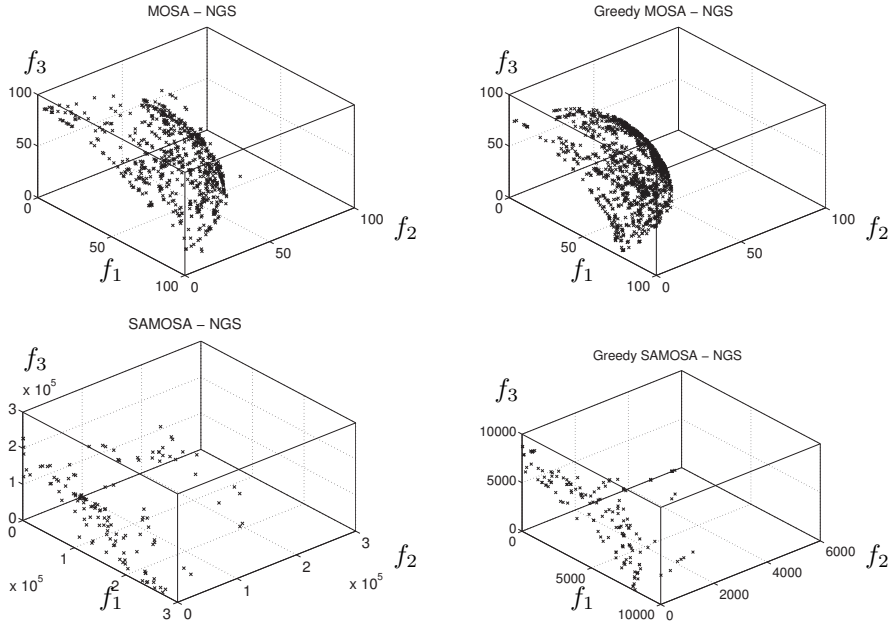

Figure 12. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on NGS(10^6) after 50000 function evaluations.

in $\mathcal{B}_{j-1}$. Thus even small *finite* perturbations to solutions in L_j are dominated. Consequently this optimisation problem cannot be solved by an infinitesimal greedy search procedure, which will inevitably become stuck on the first local front it encounters.

This reduced problem is difficult for even greedy search procedures that make finite perturbations, because in order to make a non-dominated jump from $\mathbf{x} \in \mathcal{B}_j$ to $\mathbf{x}' \in \mathcal{B}_{j-1}$ the perturbation must be large enough to jump over the part of $\mathcal{B}_{j-1}$ dominated by $\mathbf{x}$. In the simplest case when the perturbation is to x_1 solely, then the perturbation must jump one of the stripes depicted in Figure 11, that is the magnitude of the perturbation must be greater than $1/Q$. As (18) suggests, it is possible jump from $\mathcal{B}_j$ to $\mathcal{B}_{j-1}$ by perturbing x_1 and x_2 simultaneously (so that the perturbation is not radial in objective space), but in all cases a large perturbation is required to make progress towards the Pareto set.

The problem defined by (16) is generalised to a larger number of decision variables and any number of objectives by the NGS problem which defines a mapping $\mathbf{f}(\mathbf{x}) : [0,1]^P \mapsto \{\mathbf{y} : 1 \leq \|\mathbf{y}\| \leq Q + 3/2; y_i \geq 0,\ i = 1, \ldots, D\}$, as follows:

$$s^2 = \sum_{i=1}^{D} x_i^2, \tag{19a}$$

$$r = 2Q \left[\frac{1}{P-D} \sum_{i=D+1}^{P} x_i^2 \right]^{\frac{1}{2}}, \tag{19b}$$

$$R(r) = (r - \lfloor r \rfloor + 1 - \text{remainder}(\lfloor r \rfloor, 2))/2 + 1 + \lfloor \frac{r}{2} \rfloor, \tag{19c}$$

$$f_i(\mathbf{x}) = R\frac{x_i}{s}. \tag{19d}$$

Note that the discontinuous function $R(r)$ in NGS is the same function as in the reduced problem (16a). In a similar manner to the reduced problem the first D decision variables control the distance of $\mathbf{f}(\mathbf{x})$ from the Pareto front, while the remaining $P - D$ determine its angular location in objective space. The parameter Q controls the number of fronts: there are $\lceil Q/2 \rceil$ fronts arranged as concentric shells with integer radii. Regions in decision space associated with each front are also concentric annuli each of width $\sqrt{P-D}/Q$. The global Pareto front is the shell of radius 1, which corresponds to the last $P - D$ decision variables all being equal to $1/(2Q)$.

In an exactly analogous manner to the reduced problem, the NGS problem is not greedy searchable using

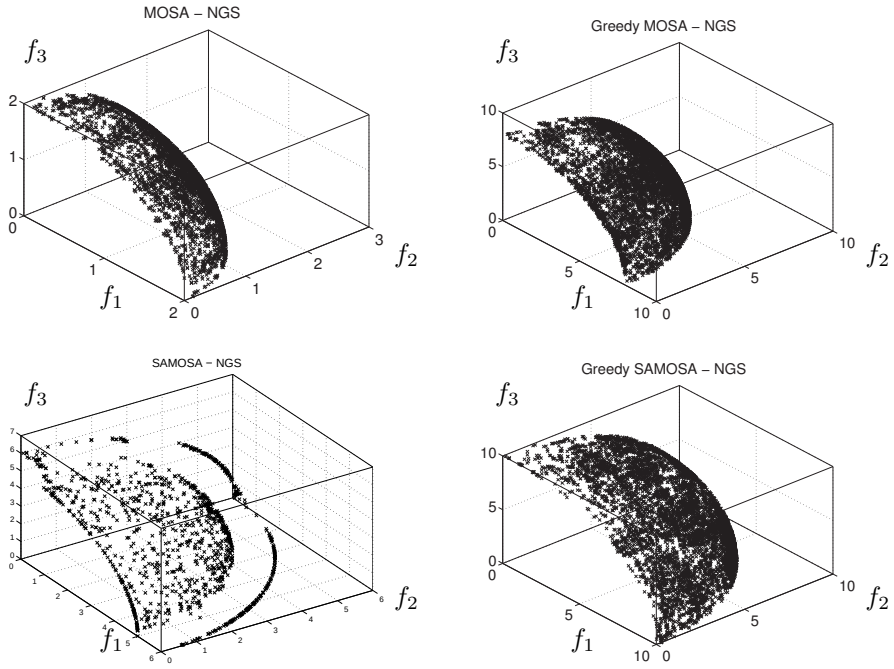

Figure 13. Archives of the median annealer run of MOSA, MOSA0, SAMOSA and SAMOSA0 on NGS(10) after 50000 function evaluations.

infinitesimal perturbations. Of course a greedy optimiser using finite perturbations will be able to locate the global Pareto front by making large perturbations from one region, over a local Pareto front, to a dominating region. Nonetheless NGS presents difficulties for procedures which perturb either a single decision variable or a few variables at a time. Suppose that m variables are perturbed simultaneously. Then, noting that $0 \leq x_i \leq 1$, the largest movement $\|\mathbf{x} - \mathbf{x}'\|$ possible is $\sqrt{m}$. Since the smallest movement required to escape a region 'behind' a local front is half the region width, namely $\sqrt{P-D}/(2Q)$, setting $\sqrt{m} < \sqrt{P-D}/(2Q)$ ensures that non-greedy moves are needed for an m-variable perturbation algorithm to reach the Pareto front. While it is initially possible to bypass fronts by performing several very large perturbations to some variables, when each variable becomes close to the optimal value it is necessary to make increasingly accurate moves to overcome the local fronts, because the annular regions of decision space behind local fronts decrease in volume as $\mathcal{P}$ is approached. A greedy algorithm is thus reduced to what is effectively an exhaustive search, while an exploratory algorithm may move between regions.

We note that, somewhat counter-intuitively, NGS with large Q and many fronts is expected to be easier than NGS with small Q, because the distance in decision space between fronts decreases as Q increases, making it easier for finite perturbations to jump over local fronts. Since the number of fronts has a great effect upon the properties of the NGS problem, it is interesting to present results on two instances, both of which are configured so that it is possible (although not necessarily likely) for the algorithms to generate perturbations which escape local fronts: the first, NGS(10^6), has $Q = 10^6$ while the second, NGS(10), has $Q = 10$. Both NGS(10^6) and NGS(10) have $P = 20$ decision variables and $D = 3$ objectives. The NGS(10^6) problem provides a landscape which is easy for a greedy optimiser to traverse, since even very small perturbations to a single decision variable can move a solution across many basins without impediment. The NGS(10) problem provides a landscape which is difficult (although not impossible) for a greedy optimiser to traverse, since very large perturbations are required to move a solution from a local optimum, over the surrounding region of greater energy, to a dominating point.

A single variable perturbation is used, and in both formulations $\sqrt{m} > \frac{\sqrt{P-D}}{2Q}$, so it is feasible to optimise the problems without resorting to exploratory moves.

Median results over 20 runs for both MOSA and SAMOSA, both with an annealing schedule[6] and fixed

[6]As earlier, all epochs were of equal length $L_k = 100$ and $T_k = \beta^k T_0$, with β chosen so that $T_k = 10^{-5}$ after approximately two thirds of the function evaluations are completed. Perturbations were drawn from a Laplacian distribution with width parameter 0.1.

at $T = 0$ are shown in Figures 12 and 13. The results for NGS(10^6), in Figure 12, show that the greedy algorithms (the annealers with $T = 0$) out-performed their corresponding non-greedy counterparts (all algorithms were initialised with a solution residing on the local front with radius approximately 10^6). MOSA and MOSA0 dramatically out-perform SAMOSA and SAMOSA0, in common with the results for the DTLZ3 problem; this is again probably because the large number of solutions available for perturbation in the SAMOSA state means that it is likely for similar moves to be made to several solutions in the set, pushing individual solutions forwards while not moving the state as a whole. Figure 13 shows the results for the algorithms on the NGS(10) problem, for which the initial solution for each algorithm resided upon a local front with radius 10. On this problem it can be seen that the greedy algorithms have difficulty leaving the initial front, with (on the median run) SAMOSA0 being unable to escape the first local front and MOSA0 being able to skip a single front. In contrast, the non-greedy algorithms were able to jump several fronts on the median run; on 4 of 20 runs MOSA was also able to converge to the true front, although SAMOSA did not reach it.
These results demonstrate the existence of problems for which it is preferable to use a non-greedy technique to converge upon the true front, even though it is possible to eventually optimise them with a greedy algorithm, and that, in the case of the NGS test problem proposed here, multi-objective simulated annealing techniques provide a method for optimisation.

6 Conclusions

This paper has explored the relationship between greedy and non-greedy optimisation algorithms based on simulated annealing with set and single solution states, and between greedy and non-greedy searchable problems.
A novel set-based multi-objective simulated annealer (SAMOSA) which stores a non-dominated set of solutions as the state was introduced. While simulated annealing requires a current state, which ultimately approximates the desired optimal solution, it is not clear whether this state should represent a single solution (as with other multi-objective simulated annealing techniques, including MOSA) or a set of solutions; as the desired result of the optimisation is a set of solutions, maintaining a set of solutions as the state is appealing (and this approach is used in many other optimisation techniques such as genetic algorithms and many evolution strategies). Comparisons between SAMOSA and MOSA showed two traits: Set-based state slowed convergence to the true Pareto front as measured by the distance of solutions from the true front; but also that, for problems with highly non-linear mappings from the region representing the true front in decision variable space to the front in objective space such as DTLZ4, methods such as Uniselect can be used with set-based annealers resulting in a considerably more uniform distribution of solutions across the true front. The slower convergence can be explained as being inherent to set-based methods of this nature, although profoundly different set-based approaches may not have this characteristic.
The use of the dominance relation to compare solutions is attractive as it does not use metric information in objective space, in contrast to both weighted sum and dominated volume approaches. Nonetheless, without knowledge of the true Pareto front, dominance does not provide an absolute energy for the state and both the MOSA and SAMOSA methods incorporate a heuristic energy *difference*. Unfortunately, this poses serious obstacles to adapting proofs of convergence for scalar simulated annealing.
The VOLMOSA algorithm exhibits performance comparable to SAMOSA on the two-objective formulations of the DTLZ 1-4 problems. However, we note that the performance of SAMOSA is considerably better on the two-objective DTLZ problems than on the three-objective formulations, yielding results both converging close to and uniformly covering the true Pareto front, and suggesting that the two-objective DTLZ formulations are significantly easier than the three-objective variants.
Empirical results on the DTLZ suite of test problems suggest that greedy searchable problems are a larger subset of all problems than might be expected. While it has previously been thought (Deb et al., 2001, 2002b) that local fronts in a problem make it difficult for greedy optimisers to converge to the true Pareto front, as they will become stuck upon the local fronts, it seems that this is not necessarily true. Additionally, while it is intuitively true that providing more local fronts makes it increasingly difficult for a greedy optimiser to converge, as it provides more locations in which one may become trapped, the opposite effect may be seen if increasing the number of fronts decreases their size in decision space, thereby increasing the chance of a perturbation being made which escapes them. Depending upon the

configuration of decision space, it is even possible for local fronts in objective space to provide no obstacle to convergence at all.

Our investigation into the characteristics of multi-objective problems shows that there are two properties which determine whether a problem is optimisable by a greedy search algorithm. The first is whether a series of infinitesimally small perturbations can be made which move a solution through decision space without being dominated by local fronts. The second is the size of perturbation required to escape a local front. When perturbations cannot avoid local fronts, and the perturbation required to escape is larger than the maximum perturbation possible with the applied scheme, it is impossible for a greedy algorithm to optimise the problem.

The introduction of a non-greedy-searchable problem, NGS, results in dramatically different performance from that on the DTLZ problems, with the greedy algorithms performing considerably worse than their exploratory counterparts. The NGS problem exhibits local fronts which must be surmounted or tunnelled through for solutions to reach the global Pareto front. The problem may be tuned in difficulty for greedy optimisers by adjusting the number (and therefore size) of these regions. A configuration of the problem with many local fronts, mapping from many small regions of decision space, allows the greedy optimisers to out-perform the corresponding exploratory algorithms, as the non-greedy moves slow, rather than aid, convergence.

Finally, we emphasise the generality of these results by pointing out the similarity of elitist evolution strategies to SAMOSA at temperature zero and algorithms like PAES to MOSA at temperature zero. Our results suggest that in many cases the characteristics of the multi-objective problem do not require exploratory steps for locating the global front, and we look forward to a more complete characterisation of 'typical' real multi-objective optimisation problems.

References

Bader, J. and Zitzler, E. (2008). HypE: An algorithm for fast hypervolume-based many-objective optimization. Technical Report 286, Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland.

Bandyopadhyay, S., Saha, S., Maulik, U., and Deb, K. (2008). A simulated annealing-based multiobjective optimization algorithm: AMOSA. *IEEE Trans on Evolutionary Computation*, 12(3):269–283.

Beume, N. and Rudolph, G. (2006). Faster S-metric calculation by considering dominated hypervolume as Klee's measure prolbem. In Kovalerchuk, B., editor, *Proceedings of the Second IASTED Conference on Computational Intelligence*, pages 231–236, Anaheim. Acta Press.

Bradstreet, L., While, L., and Barone, L. (2006). Maximising hypervolume for selection in multi-objective evolutionary algorithms. In *Proceedings of the 2006 IEEE Congress on Evolutionary Computation*, pages 6208–6215.

Bradstreet, L., While, L., and Barone, L. (2007). Incrementally maximising hypervolume for selection in multi-objective evolutionary algorithms. In *Proceedings of the 2007 IEEE Congress on Evolutionary Computation*, pages 3203–3210.

Bradstreet, L., While, L., and Barone, L. (2008). A fast incremental hypervolume algorithm. *IEEE Trans on Evolutionary Computation*, pages 1–10. To appear; available online, DOI: 10.1109/TEVC.2008.919001.

Czyżak, P. and Jaszkiewicz, A. (1998). Pareto simulated annealing – a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis*, 7:34–47.

Das, I. and Dennis, J. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14(1):63–69.

Deb, K. (1999). Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation*, 7(3):205–230.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002a). Fast and elitist multiobjective genetic algorithm: NSGA–II. *IEEE Trans on Evolutionary Computation*, 6(2):182–197.

Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. (2001). Scalable multi-objective optimization test problems. Technical Report 112, Institute für Technische Informatik und Kommunikationsnetze, ETH Zurich.

Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. (2002b). Scalable multi-objective optimization test problems.

In *Congress on Evolutionary Computation (CEC'2002)*, volume 1, pages 825–830, Piscataway, New Jersey. IEEE Service Center.

Emmerich, M., Beume, N., and Naujoks, B. (2005). An EMO algorithm using the hypervolume measure as selection criterion. In Coello Coello, C. A., editor, *Proc. Evolutionary Multi-Criterion Optimization: 3rd Int'l Conf. (EMO 2005)*, volume 3410 of *LNCS*, pages 62–76, Berlin. Springer.

Everson, R. M. and Fieldsend, J. E. (2006). Multi-objective optimisation of safety related systems: An application to short term conflict alert. *IEEE Trans on Evolutionary Computation*, 10(2):187–198.

Fieldsend, J. E., Everson, R. M., and Singh, S. (2003). Using unconstrained elite archives for multi-objective optimisation. *IEEE Trans on Evolutionary Computation*, 7(3):305–323.

Fieldsend, J. E. and Singh, S. (2002). Pareto multi-objective non-linear regression modelling to aid CAPM analogous forecasting. In *Proceedings of the 2002 IEEE International Joint Conference on Neural Networks*, pages 388–393, Hawaii, May 12-17. IEEE Press.

Fleischer, M. (2003). The measure of Pareto optima: Applications to multi-objective metaheuristics. In *Evolutionary Multi-Criterion Optimization, Second International Conference, EMO2003*, volume 2632 of *Lecture Notes in Computer Science*, pages 519–533. Springer.

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

Fonseca, C. M., Paquete, L., and López-Ibáñez, M. (2006). An improved dimension-sweep algorithm for the hypervolume indicator. In *IEEE Congress on Evolutionary Computation*, pages 1157–1163, Vancouver, Canada.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 6:721–741.

Hapke, M., Jaszkiewicz, A., and Slowinski, R. (2000). Pareto simulated annealing for fuzzy multi-objective combinatorial optimization. *Journal of Heuristics*, 6(3):329–345.

Huband, S., Barone, L., While, L., and Hingston, P. (2005). A scalable multi-objective test problem toolkit. In *Evolutionary Multi-Criterion Optimization: Third International Conference, EMO 2005*, volume 3410 of *Lecture Notes in Computer Science*, pages 280–294. Springer-Verlag. Corrected version at `http://www.wfg.csse.uwa.edu.au/publications/WFG2005a_corrected.pdf`.

Ingber, L. (1993). Simulated annealing: Practice versus theory. *Mathematical Computation and Modelling*, 18:29–57.

Jaszkiewicz, A. (2001). Comparison of local search-based metaheuristics on the multiple objective knapsack problem. *Foundations of Computer and Decision Sciences*, 26(1):99–120.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220:671–680.

Knowles, J. (2002). *Local-search and hybrid evolutionary algorithms for Pareto optimization*. PhD thesis, University of Reading, Reading, UK.

Knowles, J. and Corne, D. (1999). The Pareto Archived Evolution Strategy: A new baseline algorithm for Pareto multiobjective optimisation. In *Proceedings of the 1999 Congress on Evolutionary Computation*, pages 98–105, Piscataway, NJ. IEEE Service Center.

Laumanns, M., Thiele, L., Zitzler, E., Welzl, E., and Deb, K. (2002). Running time analysis of multi-objective evolutionary algorithms on a simple discrete optimization problem. In *Parallel Problem Solving from Nature—PPSN VII*, Lecture Notes in Computer Science, pages 44–53.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092.

Nam, D. K. and Park, C. H. (2000). Multiobjective simulated annealing: a comparative study to evolutionary algorithms. *Internation Journal of Fuzzy Systems*, 2(2):87–97.

Naujoks, B., Beume, N., and Emmerich, M. (2005). Multi-objective optimisation using S-metric selection: Application to three-dimensional solution spaces. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, pages 1282–1289.

Salamon, P., Sibani, P., and Frost, R. (2002). *Facts, Conjectures, and Improvements for Simulated Annealing*. Number 7 in Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics.

Schuur, P. (1997). Classification of acceptance criteria for the simulated annealing algorithm. *Mathematics of Operations Research*, 22(2):266–275.

Serafini, P. (1994). Simulated annealing for multiobjective optimization problems. In *Multiple criteria decision making. Expand and enrich the domains of thinking and application*, pages 283–292.

Smith, K. I. (2006). *A study of simulated annealing techniques for multi-objective optimisation*. PhD thesis, The University of Exeter, Exeter, UK.

Smith, K. I., Everson, R. M., and Fieldsend, J. E. (2004). Dominance measures for multi-objective simulated annealing. In *Proceedings of Congress on Evolutionary Computation*, pages 23–30.

Smith, K. I., Everson, R. M., Fieldsend, J. E., Misra, R., and Murphy, C. (2008). Dominance-based multiobjective simulated annealing. *IEEE Trans on Evolutionary Computation*, 12(3):323–343.

Suppapitnarm, A., Seffen, K. A., Parks, G. T., and Clarkson, P. J. (2000). A simulated annealing algorithm for multiobjective optimization. *Engineering Optimization*, 33:59–85.

Tuyttens, D., Teghem, J., and El-Sherbeny, N. (2003). A particular multiobjective vehicle routing problem solved by simulated annealing. In *Metaheuristics for multiobjective optimisation*, volume 535 of *Lecture notes in economics and mathematical systems*, pages 133–152. Springer.

Ulungu, E. L., Teghem, J., Fortemps, P., and Tuyttens, D. (1999). MOSA method: a tool for solving multiobjective combinatorial decision problems. *Journal of multi-criteria decision analysis*, 8:221–236.

Wagner, T., Beume, N., and Naujoks, B. (2007). Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In *Evolutionary Multi-Criterion Optimization, EMO2007*, pages 742–756.

Zitzler, E. and Künzli, S. (2004). Indicator-based selection in multiobjective search. In *Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, Heidelberg, Germany. Springer.

Zitzler, E. and Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans on Evolutionary Computation*, 3(4):257–271.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Grunert da Fonseca, V. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans on Evolutionary Computation*, 7(2):117–132.