IPSJ Transactions on Mathematical Modeling and Its Applications Vol. 2 No. 3 39–53 (Dec. 2009)

Regular Paper

ϵ-Ranking for Effective Many Objective Optimization on MNK-Landscapes

Hernán Aguirre[†1,†2] and Kiyoshi Tanaka[†2]

This work proposes a method to enhance selection of multiobjective evolutionary algorithms aiming to improve their performance on *many* objective optimization problems. The proposed method uses a randomized sampling procedure combined with ϵ-dominance to fine grain the ranking of solutions after they have been ranked by Pareto dominance. The sampling procedure chooses a subset of initially equal ranked solutions to give them selective advantage, favoring a good distribution of the sample based on dominance regions wider than conventional Pareto dominance. We enhance NSGA-II with the proposed method and analyze its performance on a wide range of non-linear problems using MNK-Landscapes with up to $M = 10$ objectives. Experimental results show that convergence and diversity of the solutions found can improve remarkably on $3 \leq M \leq 10$ objective problems.

1. Introduction

Multiobjective evolutionary algorithms (MOEAs) [1),2)] optimize simultaneously two or more objective functions, aiming to find a set of compromised Pareto optimal solutions in a single run of the algorithm. Most state of the art MOEAs use Pareto dominance within the selection procedure of the algorithm to rank solutions. Selection based on Pareto dominance is thought to be effective for problems with convex and non-convex fronts and has been successfully applied in two and three objectives problems.

Recently, there is a growing interest on applying MOEAs to solve *many* objectives optimization problems [3)–9)], where the number of objectives to optimize is more than three. However, MOEAs that use selection based on Pareto dominance do not scale-up well on *many* objective problems. It is known that the number of Pareto non-dominated solutions increase substantially as we increase the number of objectives of the problem [3)]. Consequently, ranking by Pareto dominance becomes coarser and too many solutions are assigned highest rank. This affects the effectiveness of selection, severely deteriorating the performance of MOEAs [4)–7)].

In this work, we propose a method to enhance selection of MOEAs aiming to improve their performance on *many* objective optimization problems. The proposed method uses a randomized sampling procedure combined with ϵ-dominance to fine grain the ranking of solutions after they have been ranked by Pareto dominance. The sampling procedure chooses a subset of initially equal ranked solutions to give them selective advantage, favoring a good distribution of the sample based on dominance regions wider than conventional Pareto dominance. Basically, sampled solutions keep their initial rank and solutions located within the virtually expanded dominance region of the sampled solutions are demoted to an inferior rank. Thus, the proposed ranking method increases selection probabilities of some of the solutions, while trying to keep a uniform search effort towards the different zones of objective space represented in the actual population.

We verify the effectiveness of the proposed method using NSGA-II's framework. NSGA-II [10)] is a widely referenced algorithm that has been shown to perform relatively well on two and three objective problems, but known not to scale-up well on *many* objective problems. We enhance NSGA-II with ϵ-ranking and analyze its performance on a wide range of combinatorial non-linear problems using MNK-Landscapes [3),5)]. An MNK-Landscape is a scalable model of epistatic interactions for multiobjective combinatorial problems and it is useful to understand the fundamental search mechanisms of the algorithms and their scalability by varying the non-linear complexity K, number of objectives M, and size of the search space 2^N of the problem. In our experiments we use problems with $2 \leq M \leq 10$ objectives, $N = 100$ bits, and $0 \leq K \leq 50$ epistatic, non-linear, interactions per bit. In addition, we compare results and computational time by the proposed method, conventional NSGA-II, Subvector Dominance Assignment (SVDOM), and Epsilon Dominance Assignment (EPSDOM). SVDOM and EPSDOM are methods recently proposed to enhance NSGA-II for many-objective optimization [8)].

†1 International Young Researcher Empowerment Center, Shinshu University
†2 Faculty of Engineering, Shinshu University

2. Multiobjective Optimization Concepts and Definitions

Let us consider, without loss of generality, a maximization multiobjective problem with M objectives:

$$maximize \quad \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x})) \tag{1}$$

where $\boldsymbol{x} \in \mathcal{S}$ is a solution vector in the feasible solution space $\mathcal{S}$, and $f_1(\cdot)$, $f_2(\cdot)$, $\cdots$, $f_M(\cdot)$ the M objectives to be maximized.

In decision making, one dimensional comparison and Pareto optimality are two popular methods used to choose a solution once a set of Pareto optimal solutions has been found. Yu [11] showed that these two methods are two extreme cases in the entire domain of domination structures and that there are infinity valid methods lying between them, which suitability depends on how much information is known on the decision maker's preferences. Within the EMO community, these other domination structures are also known as relaxed forms of Pareto dominance and one method to implement them is ϵ-dominance [12]. In the following we define Pareto dominance and ϵ-dominance, two concepts that are of special relevance to this work.

Pareto dominance. A solution $\boldsymbol{x}$ Pareto dominates other solution $\boldsymbol{y}$, denoted by $\boldsymbol{f}(\boldsymbol{x}) \succeq \boldsymbol{f}(\boldsymbol{y})$, if the following conditions are satisfied:

$$\begin{aligned} &\forall m \in \{1, \cdots, M\}\ f_m(\boldsymbol{x}) \geq f_m(\boldsymbol{y}) \quad \wedge \\ &\exists m \in \{1, \cdots, M\}\ f_m(\boldsymbol{x}) > f_m(\boldsymbol{y}). \end{aligned} \tag{2}$$

ϵ-dominance. A solution $\boldsymbol{x}$ ϵ-dominates other solution $\boldsymbol{y}$, denoted by $\boldsymbol{f}(\boldsymbol{x}) \succeq^{\epsilon} \boldsymbol{f}(\boldsymbol{y})$, if the following conditions are satisfied:

$$\begin{aligned} &\forall m \in \{1, \cdots, M\}\ (1+\epsilon) f_m(\boldsymbol{x}) \geq f_m(\boldsymbol{y}) \quad \wedge \\ &\exists m \in \{1, \cdots, M\}\ (1+\epsilon) f_m(\boldsymbol{x}) > f_m(\boldsymbol{y}), \end{aligned} \tag{3}$$

where $\epsilon > 0.0$.

Figure 1 illustrates for one solution its Pareto dominance region and its ϵ-dominance region.

3. Method

The core of the proposed method is a randomized ϵ-sampling procedure that favors a good distribution of solutions based on dominance regions wider than conventional Pareto dominance. In the following, we first explain ϵ-sampling and then ϵ-ranking to fine grain ranking of solutions. Finally, we explain the enhanced NSGA-II [10].

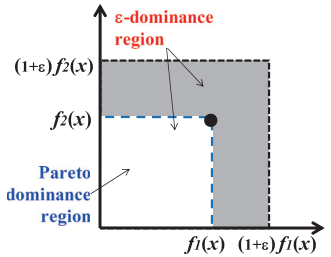


Fig. 1 Pareto dominance and ϵ-dominance.

3.1 ϵ-Sampling

ϵ-Sampling assumes that there is a set of equally ranked solutions from which a subset of them should be chosen to be given selective advantage in order to proceed further with the evolutionary search. That is, ϵ-sampling acts as a decision making procedure, not to find a final solution, but to help selection of the evolutionary algorithm. Hence, the sampling heuristic must reflect criteria that favor an effective search. Here, the sample of solutions to be given selective advantage are obtained with the following criteria,

- Extreme solutions are always part of the sample.
- Each (not extreme) sampled solution is the sole sampled representative of its area of influence. The area of influence of the sampled solutions is determined by a domination region wider than Pareto dominance, i.e., ϵ-dominance.
- Sampling of (not extreme) solutions follows a random schedule.

The first criterion tries to push the search towards the optimum values of each fitness function, aiming to find non-dominated solutions in a wide area of objective space. The second criterion assures that only one solution in a given zone of objective space is given higher rank, trying to distribute the search effort more or less uniformly among the different zones represented in the actual population. The third criterion dynamically establishes the zones that are represented in the sample. Also, in the case that there are several solutions within each zone, it

increases the likelihood that the sampled solutions that will be given higher rank are different from one generation to the next, increasing the possibility of exploring wider areas of objective and variable space. This is an important issue in elitist algorithms, like most state of the art MOEAs. Summarizing, the proposed sampling method is a decision making procedure used by the algorithm in order to increase selection probabilities of some of the solutions, trying to keep a uniform search effort towards the different zones of objective space.

Procedure 1 illustrates the algorithm of the proposed ϵ-sampling method. Let us denote $\mathcal{A}$ the set of solutions that have been assigned the same rank based on conventional Pareto dominance, for example by applying non-domination sorting [10]. ϵ-Sampling returns the sampled solutions $\mathcal{S} \subset \mathcal{A}$ that will be given selective advantage as well as the set of solutions $\mathcal{D}^\epsilon$ to be demoted. See that extreme solutions are the first to be assigned to the sample $\mathcal{S}$ (lines 1, 2). Then, one by one, solutions are randomly chosen and included in $\mathcal{S}$ (lines 6–8), whereas

Procedure 1 ϵ-sampling (ϵ, $\mathcal{A}$, $\mathcal{S}$, $\mathcal{D}^\epsilon$)

Input: ϵ-dominance factor ϵ and a set of equal ranked solutions $\mathcal{A}$

Output: $\mathcal{S}$ and $\mathcal{D}^\epsilon$ ($\mathcal{S} \cup \mathcal{D}^\epsilon = \mathcal{A}$). $\mathcal{S}$ contains extreme and ϵ-non-dominated solutions, whereas $\mathcal{D}^\epsilon$ contains ϵ-dominated solutions

1: $\mathcal{X} \leftarrow \{x \in \mathcal{A} \mid f_m(x) = \max(f_m(\cdot)),\ m = 1, 2, \cdots, M\}$
2: $\mathcal{S} \leftarrow \mathcal{X}$
3: $\mathcal{A} \leftarrow \mathcal{A} \setminus \mathcal{X}$
4: $\mathcal{D}^\epsilon \leftarrow \emptyset$
5: **while** $\mathcal{A} \neq \emptyset$ **do**
6: $r \leftarrow rand(),\ 1 \leq r \leq |\mathcal{A}|$
7: $z \leftarrow r$-th solution $\in \mathcal{A}$
8: $\mathcal{S} \leftarrow \mathcal{S} \cup \{z\}$
9: $\mathcal{Y} \leftarrow \{y \in \mathcal{A} \mid z \succeq^\epsilon y, z \neq y\}$
10: $\mathcal{D}^\epsilon \leftarrow \mathcal{D}^\epsilon \cup \mathcal{Y}$
11: $\mathcal{A} \leftarrow \mathcal{A} \setminus \{\{z\} \cup \mathcal{Y}\}$
12: **end while**
13: **return**

solutions that lie in the wider domination region of the randomly picked solution are assigned to $\mathcal{D}^\epsilon$ (lines 9, 10).

3.2 ϵ-Ranking

The ϵ-sampling procedure works on a set of equally ranked solutions, however within a population there could be several sets of such solutions (each set with a different rank). Here, we explain ϵ-ranking for NSGA-II to re-rank all possible sets of equally ranked solutions using the ϵ-sampling method.

In NSGA-II, a non-domination sorting procedure is applied to the joined population of parents and offspring to classify solutions in fronts of non-dominated solutions. Let us denote $\mathcal{Z}$ the set of solution we want to classify. The first front $\mathcal{F}_1$ is obtained from $\mathcal{Z}$ and corresponds to the set of Pareto optimal solutions in $\mathcal{Z}$. Let us denote this set as $\mathcal{POS}_1$. The subsequent fronts $\mathcal{F}_i$, $i > 1$, contain lower level non-dominated solutions and are obtained by disregarding solutions corresponding to the previously classified fronts, i.e., $\mathcal{F}_i$, $i > 1$, is obtained from the set $\mathcal{Z} - \bigcup_{k=1}^{i-1} \mathcal{POS}_k$. Solutions in each front are assigned the same non-domination rank, equal to the front number they belong to.

ϵ-Ranking in NSGA-II is applied at each generation after non-domination sorting to reclassify the fronts $\mathcal{F}_i$ ($i = 1, \cdots, N_F$) found by NSGA-II. **Procedure 2** describes the ϵ-ranking method for NSGA-II. See that the reclassified front $\mathcal{F}_j^\epsilon$ ($j = 1, \cdots, N_F^\epsilon$) now contains only the sample of solutions $\mathcal{S} \subset \mathcal{F}_i$ found by ϵ-sampling (lines 9, 10). Also, see that solutions $\mathcal{D}^\epsilon$, which are not part of the sample (line 9) are demoted by joining them with solutions of an inferior front in the next iteration of the loop (line 4). Thus, $\mathcal{F}_1^\epsilon$ contains some of the solutions initially ranked first, but $\mathcal{F}_j^\epsilon$, $j > 1$, can contain solutions that initially were ranked in different fronts. This gives chance to lateral diversity present in the initial ranking of solutions and can punish highly crowded solutions even if they are initially ranked first by conventional Pareto dominance. **Figure 2** illustrates the application of ϵ-sampling on the first front $\mathcal{F}_1$ and on the second front $\mathcal{F}_2$ joined with the demoted solutions $\mathcal{D}^\epsilon$ from $\mathcal{F}_1$. A number close to the solutions represents the random schedule in which solutions are sampled (0 means extreme solutions, which are all selected at once).

Truncation to obtain the new parent population is applied after re-classifying fronts with ϵ-ranking. Many demoted solutions would still be part of the new

Procedure 2 ϵ-ranking (ϵ, $\mathcal{F}$, $\mathcal{F}^\epsilon$)

Input: ϵ-dominance factor ϵ and solutions $\mathcal{F}$ classified in fronts $\mathcal{F}_i$ ($i = 1, \cdots, N_F$) by non-domination sorting

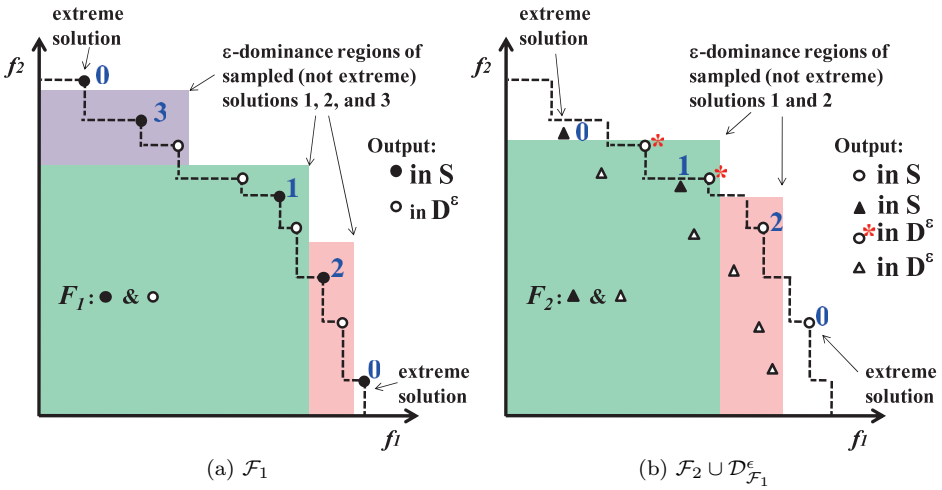
Output: $\mathcal{F}^\epsilon$, solutions re-classified in fronts $\mathcal{F}^\epsilon_j$ ($j = 1, \cdots, N^\epsilon_F$) after ϵ-sampling

1: $\mathcal{D}^\epsilon \leftarrow \emptyset$, $i \leftarrow 1$, $j \leftarrow 1$
2: **repeat**
3: **if** $i \leq N_F$ **then**
4: $\mathcal{A} \leftarrow \mathcal{F}_i \cup \mathcal{D}^\epsilon$
5: $i \leftarrow i + 1$
6: **else**
7: $\mathcal{A} \leftarrow \mathcal{D}^\epsilon$
8: **end if**
9: ϵ-sampling(ϵ, $\mathcal{A}$, $\mathcal{S}$, $\mathcal{D}^\epsilon$)
10: $\mathcal{F}^\epsilon_j \leftarrow \mathcal{S}$
11: $j \leftarrow j + 1$
12: **until** $\mathcal{D}^\epsilon = \emptyset$
13: **return**

parent population, though their probability to reproduce has been reduced instantaneously by assigning them an inferior rank. Demoted solutions, which are not dominated by offspring, could be part of the highest ranked sample in the next generation, because ϵ-sampling would be applied again in a randomized manner.

3.3 ϵ-Ranking in NSGA-II Framework

In this work, we enhance NSGA-II [10] with the proposed ϵ-ranking method as illustrated in **Procedure 3**. Note that solutions are firstly ranked based on Pareto dominance and its crowding estimated using non-domination sorting and crowding distance procedures [10], respectively (lines 4, 5). After this initial ranking, solutions are classified in sets of non-dominated solutions $\mathcal{F} = \{\mathcal{F}_i\}$ ($i = 1, 2, \cdots, N_F$). However, in *many* objective problems a large number of solutions are expected to be given the highest rank. Thus, a finer grained ranking of solutions (line 6) is created by using a ϵ-ranking procedure, i.e., ϵ-ranking



(a) $\mathcal{F}_1$ (b) $\mathcal{F}_2 \cup \mathcal{D}^\epsilon_{\mathcal{F}_1}$

Fig. 2 ϵ-sampling on (a) the first front and (b) second front joined with solutions demoted from the first.

Procedure 3 NSGA-II with ϵ-ranking

Input: ϵ-dominance factor $\epsilon > 0.0$

Output: $\mathcal{F}_1$, set of Pareto non-dominated solutions

1: $\mathcal{P} \leftarrow \emptyset$, $\mathcal{Q} \leftarrow$ *random* // initialize parents $\mathcal{P}$ and offspring $\mathcal{Q}$
2: **repeat**
3: evaluation($\mathcal{Q}$)
4: $\mathcal{F} \leftarrow$ non-domination-sorting($\mathcal{P} \cup \mathcal{Q}$) // $\mathcal{F} = \{\mathcal{F}_i\}$ ($i = 1, 2, \cdots, N_F$)
5: crowding-distance($\mathcal{F}$)
6: $\mathcal{F}^\epsilon \leftarrow \epsilon$-ranking($\epsilon$, $\mathcal{F}$) // $\mathcal{F}^\epsilon = \{\mathcal{F}^\epsilon_j\}$ ($j = 1, 2, \cdots, N^\epsilon_F$)
7: $\mathcal{P} \leftarrow$ truncation($\mathcal{F}^\epsilon$) // $|\mathcal{P}| = |\mathcal{Q}|$
8: $\mathcal{Q} \leftarrow$ recombination and mutation($\mathcal{P}$)
9: **until** termination criterion is met
10: **return** $\mathcal{F}_1$

reclassifies $\mathcal{F}$ into $\mathcal{F}^\epsilon = \{\mathcal{F}^\epsilon_j\}$ ($i = 1, 2, \cdots, N^\epsilon_F$), where $N^\epsilon_F \geq N_F$. ϵ-ranking uses parameter $\epsilon > 0.0$ to virtually extend the dominance area of the sampled solutions in order not to include closely located solutions in the sample.

The new ranking of solutions is reflected in the selection procedure of the algorithm, both during population truncation and during mating for recombination. That is, the parent population $\mathcal{P}$ is obtained by truncating $\mathcal{F}^{\epsilon}$ (line 7). Namely, groups of solutions $\mathcal{F}_j^{\epsilon}$ are assigned iteratively to $\mathcal{P}$, starting with $\mathcal{F}_1^{\epsilon}$. If $\mathcal{F}_j^{\epsilon}$ overfills $\mathcal{P}$, crowding distance calculated in $\mathcal{F}_i$ is used to choose the required number of solutions. Also, mating for recombination is carried out by binary tournaments using the new rank of solutions ($\mathcal{F}_j^{\epsilon}$) to determine the winners, breaking ties by crowding distance calculated in $\mathcal{F}_i$.

4. Test Problems, Performance Measures and Parameters

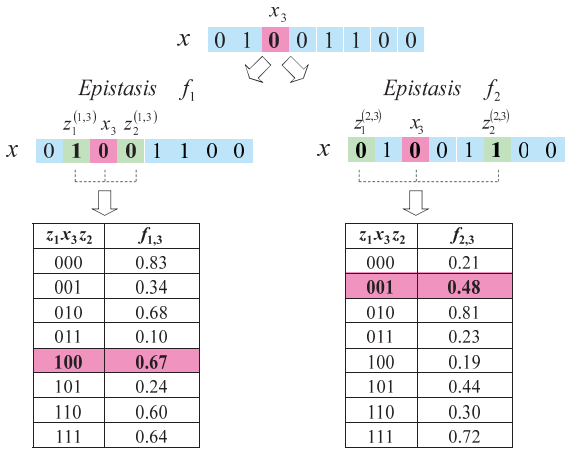
4.1 Multiobjective MNK-Landscapes

In this work we test the performance of the algorithms on multiobjective MNK-Landscapes. A multiobjective MNK-Landscape [3),5)] is defined as a vector function mapping binary strings into real numbers $\boldsymbol{f}(\cdot) = (f_1(\cdot), f_2(\cdot), \cdots, f_M(\cdot)) : \mathcal{B}^N \rightarrow \Re^M$, where M is the number of objectives, $f_i(\cdot)$ is the i-th objective function, $\mathcal{B} = \{0, 1\}$, and N is the bit string length. $\boldsymbol{K} = \{K_1, \cdots, K_M\}$ is a set of integers where K_i ($i = 1, 2, \cdots, M$) is the number of bits in the string that epistatically interact with each bit in the i-th landscape. Each $f_i(\cdot)$ can be expressed as an average of N functions as follows

$$f_i(\boldsymbol{x}) = \frac{1}{N}\sum_{j=1}^{N} f_{i,j}(x_j, z_1^{(i,j)}, z_2^{(i,j)}, \cdots, z_{K_i}^{(i,j)}) \qquad (4)$$

where $f_{i,j} : \mathcal{B}^{K_i+1} \rightarrow \Re$ gives the fitness contribution of bit x_j to $f_i(\cdot)$, and $z_1^{(i,j)}, z_2^{(i,j)}, \cdots, z_{K_i}^{(i,j)}$ are the K_i bits interacting with bit x_j in the string $\boldsymbol{x}$. The fitness contribution $f_{i,j}$ of bit x_j is a number between $[0.0, 1.0]$ drawn from a uniform distribution. Thus, each $f_i(\cdot)$ is a non-linear function of $\boldsymbol{x}$ expressed by a Kauffman's NK-Landscape model of epistatic interactions [13)]. **Figure 3** shows an example of the two fitness functions, based on different epistatic models, associated to the same bit in a two objective landscape.

For a given N, we can tune the ruggedness of the fitness function $f_i(\cdot)$ of the i-th objective by varying K_i. In the limits, $K_i = 0$ corresponds to a model in which there are no epistatic interactions and the fitness contribution from each bit value is simply additive, which yields a single peaked smooth i-th fitness landscape. On the opposite extreme, $K_i = N-1$ corresponds to a model in which each bit value is epistatically affected by all the remaining bit values yielding a maximally rugged fully random i-th fitness landscape. Varying K_i from 0 to $N-1$ gives a family of increasingly rugged multi-peaked landscapes.



$z_1 x_3 z_2$	$f_{1,3}$
000	0.83
001	0.34
010	0.68
011	0.10
100	**0.67**
101	0.24
110	0.60
111	0.64

$z_1 x_3 z_2$	$f_{2,3}$
000	0.21
001	**0.48**
010	0.81
011	0.23
100	0.19
101	0.44
110	0.30
111	0.72

Fig. 3 An example of the fitness functions $f_{1,3}(x_3, z_1^{(1,3)}, z_2^{(1,3)})$ and $f_{2,3}(x_3, z_1^{(2,3)}, z_2^{(2,3)})$ associated to bit x_3 contributing to the first objective function $f_1(\cdot)$ and second one $f_2(\cdot)$, respectively. In $f_{1,3}$ x_3 epistatically interacts with its left and right neighboring bits, $z_1^{(1,3)} = x_2$ and $z_2^{(1,3)} = x_4$. On the other hand, in $f_{2,3}$ x_3 epistatically interacts with its second bit to the left and with its third bit to the right, $z_1^{(2,3)} = x_1$ and $z_2^{(2,3)} = x_6$. ($N = 8$, $K_1 = K_2 = 2$).

In addition to M, N, and $\boldsymbol{K}$, it is also possible to specify the epistatic pattern between bit x_j and the K_i other interacting bits. That is, the distribution $D_i = \{random, nearest\ neighbor\}$ of K_i bits among N. Thus, M, N, $\boldsymbol{K} = \{K_1, K_2, \cdots, K_M\}$, and $\boldsymbol{D} = \{D_1, D_2, \cdots, D_M\}$, completely specify a multiobjective MNK-Landscape.

From a multiobjective random test problem generator standpoint [14)], desirable features of MNK-Landscapes are that the problems are easy to construct and can scale to any number of objectives M, number of bits N, and number of epistatic interactions K_i, allowing the creation of sub-classes of combinatorial non-linear problems for discrete search spaces. By varying these parameters we can analyze

the properties of the multiobjective landscapes and study the effects of the number of objectives, size of the search space, intensity of epistatic interactions, and epistatic pattern on the performance of multiobjective optimization algorithms on combinatorial discrete search spaces.

4.2 Performance Measures

In this work, we use the hypervolume $\mathcal{H}$ and coverage $\mathcal{C}$ measures [15] to evaluate and compare the performance of the algorithms. The measure $\mathcal{H}$ calculates the volume of the M-dimensional region in objective space enclosed by a set of non-dominated solutions and a dominated reference point. Let $\mathcal{A}$ be a set of non-dominated solutions. The hypervolume of $\mathcal{A}$ can be expressed as

$$\mathcal{H}(\mathcal{A}) = \cup_{i=1}^{|\mathcal{A}|}(\mathcal{V}_i - \cap_{j=1}^{i-1}\mathcal{V}_i\mathcal{V}_j) \tag{5}$$

where $\mathcal{V}_i$ is the hypervolume rendered by the point $\boldsymbol{x}_i \in \mathcal{A}$ and the reference point. In this work, the reference point is set to $[0.0, \cdots, 0.0]$. Given two sets of non-dominated solutions $\mathcal{A}$ and $\mathcal{B}$, if $\mathcal{H}(\mathcal{A}) > \mathcal{H}(\mathcal{B})$ then set $\mathcal{A}$ can be considered better on convergence and/or diversity of solutions. To calculate $\mathcal{H}$, we use Fonseca, et al. [16] algorithm, which significantly reduces computational time.

The coverage $\mathcal{C}$ measure [15] provides complementary information on convergence. Let us denote $\mathcal{A}$ and $\mathcal{B}$ the sets of non-dominated solutions found by two algorithms. $\mathcal{C}(\mathcal{A}, \mathcal{B})$ gives the fraction of solutions in $\mathcal{B}$ that are dominated at least by one solution in $\mathcal{A}$. More formally,

$$\mathcal{C}(\mathcal{A}, \mathcal{B}) = \frac{\mid \{\boldsymbol{b} \in \mathcal{B} | \exists \boldsymbol{a} \in \mathcal{A} : \boldsymbol{f}(\boldsymbol{a}) \succeq \boldsymbol{f}(\boldsymbol{b})\} \mid}{\mid \mathcal{B} \mid}. \tag{6}$$

$\mathcal{C}(\mathcal{A}, \mathcal{B}) = 1.0$ indicates that all solutions in $\mathcal{B}$ are dominated by solutions in $\mathcal{A}$, whereas $\mathcal{C}(\mathcal{A}, \mathcal{B}) = 0.0$ indicates that no solution in $\mathcal{B}$ is dominated by solutions in $\mathcal{A}$. Since usually $\mathcal{C}(\mathcal{A}, \mathcal{B}) + \mathcal{C}(\mathcal{B}, \mathcal{A}) \neq 1.0$, both $\mathcal{C}(\mathcal{A}, \mathcal{B})$ and $\mathcal{C}(\mathcal{B}, \mathcal{A})$ are required to understand the degree to which solutions of one set dominate solutions of the other set.

4.3 Parameters

In this work, we test the performance of the algorithm using MNK-Landscapes with $2 \leq M \leq 10$ objectives, $N = 100$ bits, number of epistatic interactions $K = \{0, 1, 3, 5, 10, 15, 25, 35, 50\}$ $(K_1, \cdots, K_M = K)$, and *random* epistatic patterns among bits for all objectives (D_1, $\cdots$, D_M = *random*). Results presented below show the average performance of the algorithms on 50 different problems randomly generated for each combination of M, N and K. That is, 4,500 different problems are used in total. In the plots, error bars show 95% confidence intervals on the mean.

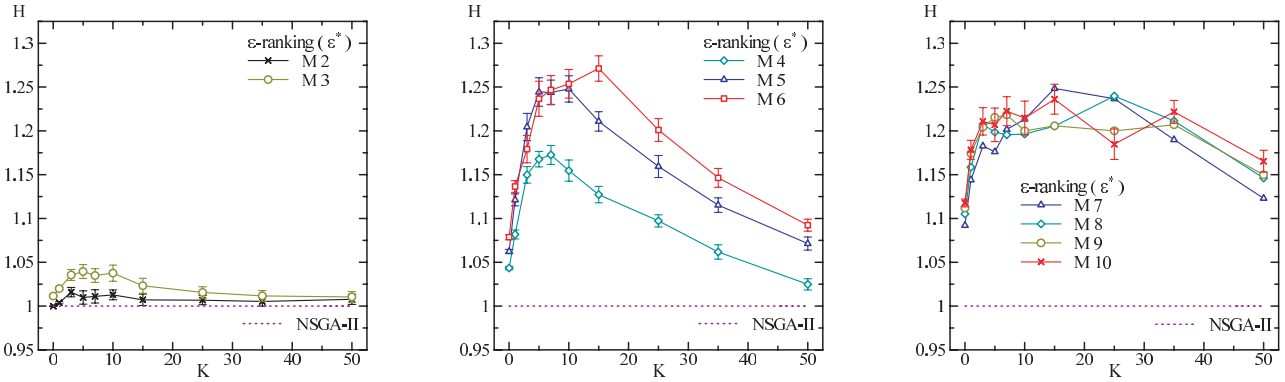
In the following sections we analyze results by NSGA-II enhanced with the proposed method (referred as ϵ-ranking for short) comparing them with results by conventional NSGA-II. ϵ-ranking and conventional NSGA-II use parent and offspring populations of size 100, two point crossover for recombination with rate $p_c = 0.6$, and bit flipping mutation with rate $p_m = 1/N$ per bit. The number of evaluations is set to 3×10^5. To have a broad picture of the effect of the wider dominations regions used within ϵ-sampling, in our study we conduct experiments varying ϵ on the range $[0.5, 10.0]$ (%) in intervals of 0.5 for $2 \leq M \leq 6$ objectives problems; whereas for $7 \leq M \leq 10$ objectives we vary ϵ on the ranges $[1.0, 10.0]$ (%) in intervals of 0.5 and $[12.5, 22.5]$ (%) in intervals of 2.5.

5. Experimental Results and Discussion

5.1 Performance by ϵ-Ranking and Conventional NSGA-II

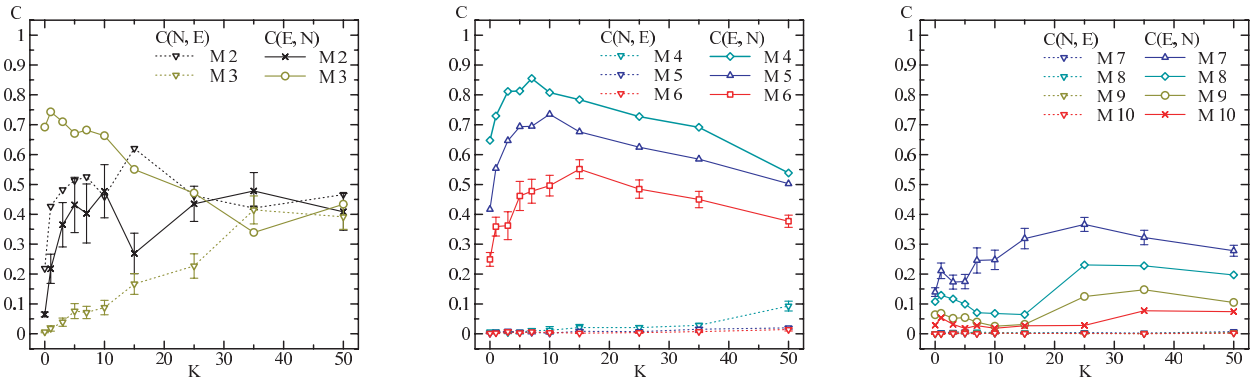
In this section, we first discuss the relative gains on performance by ϵ-ranking set with ϵ^* that achieves maximum hypervolume $\mathcal{H}$. **Figure 4** shows the average ratio $\frac{\mathcal{H}(E)}{\mathcal{H}(N)}$, where E and N denote the set of solutions found by ϵ-ranking and conventional NSGA-II, respectively. Thus, a ratio greater than 1.0 indicates better $\mathcal{H}$ by ϵ-ranking than conventional NSGA-II. As a reference, we include a horizontal line to represent the $\mathcal{H}(N)$ values normalized to 1.0. From this figure, we can see that ϵ-ranking can slightly improve $\mathcal{H}$ on problems with $M = 2$ and $M = 3$ objectives for some values of K (4% improvement or less). On the other hand, for $4 \leq M \leq 10$ objectives, the improvement on $\mathcal{H}$ is remarkable for most values of K (up to 27% improvement). Note that improvements on $\mathcal{H}$ become larger as we increase the number of objectives M from 2 to 6, whereas improvements on $\mathcal{H}$ are similar for $7 \leq M \leq 10$.

Improvements on $\mathcal{H}$ can be due to solutions with better convergence, better diversity, or both. To complement the analyzes of results on $\mathcal{H}$ we also present results using the $\mathcal{C}$ measure. **Figure 5** shows the average $\mathcal{C}$ values between conventional NSGA-II and ϵ-ranking set with ϵ^*. From this figure, we can see that



(a) $\mathcal{H}$, $M = \{2, 3\}$ (b) $\mathcal{H}$, $M = \{4, 5, 6\}$ (c) $\mathcal{H}$, $M = \{7, 8, 9, 10\}$

Fig. 4 Normalized $\mathcal{H}$, ϵ-ranking set with ϵ^* that achieves maximum $\mathcal{H}(E)$.

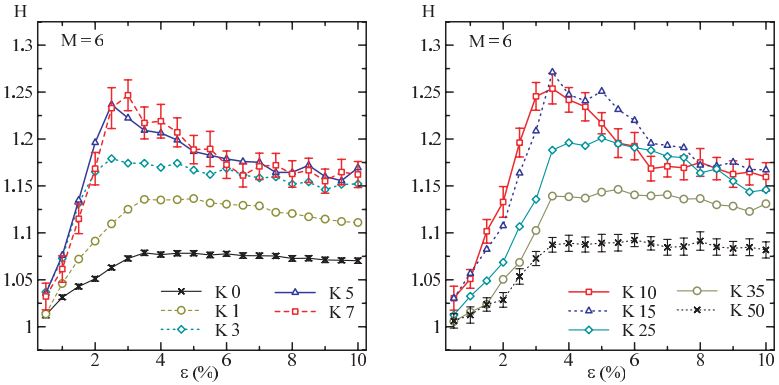


(a) $\mathcal{C}$, $M = \{2, 3\}$ (b) $\mathcal{C}$, $M = \{4, 5, 6\}$ (c) $\mathcal{C}$, $M = \{7, 8, 9, 10\}$

Fig. 5 $\mathcal{C}$ between NSGA-II and ϵ-ranking set with ϵ^* that achieves maximum $\mathcal{H}(E)$.

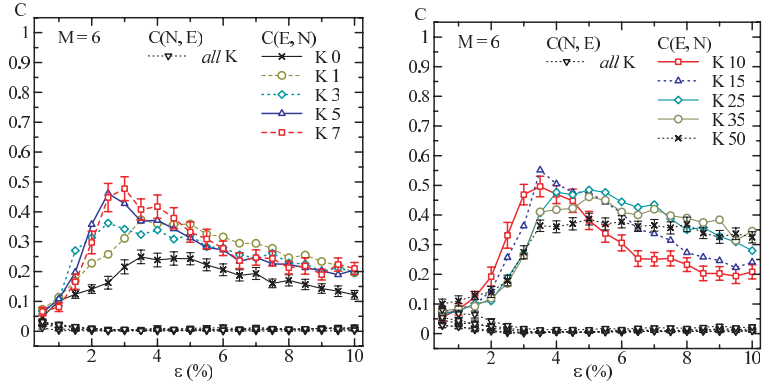
$\mathcal{C}(E, N)$ is slightly smaller than $\mathcal{C}(N, E)$ for $M = 2$ and $K \leq 10$, which means that convergence is somewhat worse by ϵ-ranking than conventional NSGA-II. Thus, the slight improvement on $\mathcal{H}$ by ϵ-ranking, observed in the same region $M = 2$ and $0 \leq K \leq 10$ as shown in Fig. 4 (a), is at the expense of a slight deterioration on convergence.

In the case of $M = 3$ objectives, we can see that $\mathcal{C}(E, N)$ is considerably greater than $\mathcal{C}(N, E)$ for $0 \leq K \leq 25$, which means that convergence is better by ϵ-ranking than conventional NSGA-II. Thus, we can conclude that a better convergence of solutions contributes to the improvement on $\mathcal{H}$ by ϵ-ranking, as shown in Fig. 4 (a) for $0 \leq K \leq 25$.



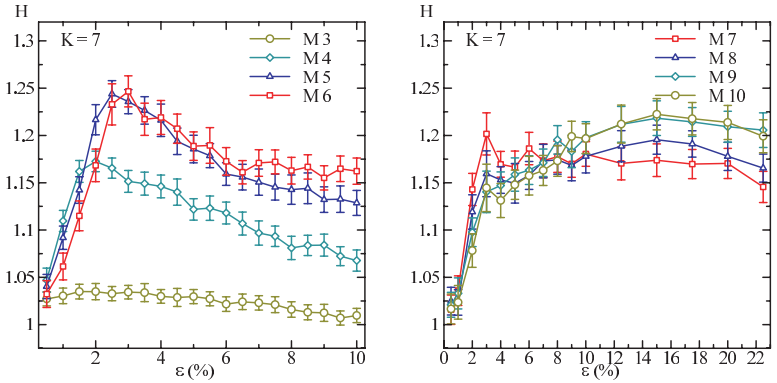
(a) $M = 6$, $K = \{0, 1, 3, 5, 7\}$ (b) $M = 6$, $K = \{10, 15, 25, 35, 50\}$

Fig. 6 Normalized $\mathcal{H}$ varying ϵ, $M = 6$, $K = \{0, 1, 3, 5, 7, 10, 15, 25, 35, 50\}$.



(a) $M = 6$, $K = \{0, 1, 3, 5, 7\}$ (b) $M = 6$, $K = \{10, 15, 25, 35, 50\}$

Fig. 7 $\mathcal{C}$ between NSGA-II and ϵ-ranking varying ϵ, $M = 6$, $K = \{0, 1, 3, 5, 7, 10, 15, 25, 35, 50\}$.



(a) $M = \{3, 4, 5, 6\}$, $K = 7$ (b) $M = \{7, 8, 9, 10\}$, $K = 7$

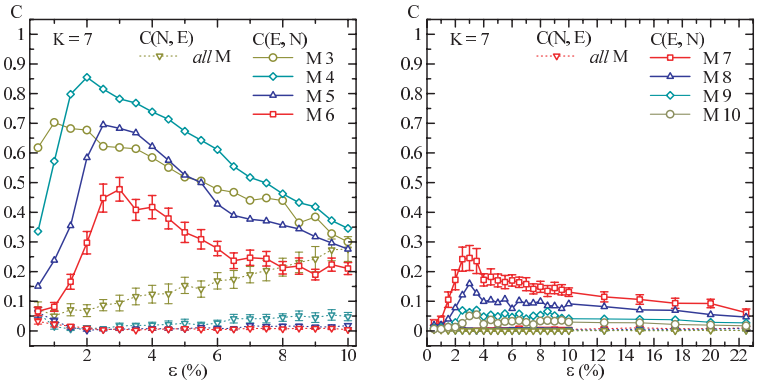
Fig. 8 Normalized $\mathcal{H}$ varying ϵ, $M = \{3, 4, 5, 6, 7, 8, 9, 10\}$, $K = 7$.

For $4 \leq M \leq 10$ a clear trend can be seen. $\mathcal{C}(N, E)$ is close to 0.0 for most K and M. This indicates that there are almost no solutions by conventional NSGA-II that dominate solutions by ϵ-ranking. On the other hand, the values of $\mathcal{C}(E, N)$ are very high for 4 objectives (in the range 0.55–0.85) and reduce gradually as we increase M up to 10 objectives (in the range 0.01–0.08). This suggests that a better convergence of solutions contributes to the increases of $\mathcal{H}$ by ϵ-ranking on $M = 4$ problems. As we increase M, gains on diversity gradually become more significant than gains on convergence as the reason for the remarkable improvement of $\mathcal{H}$ on $5 \leq M \leq 10$.

5.2 Effect of ϵ

Next, we discuss the effects on performance of parameter ϵ. **Figures 6** and **7** show, for $M = 6$ objectives and various K, the ratio $\frac{\mathcal{H}(E)}{\mathcal{H}(N)}$ and the $\mathcal{C}$ measure between NSGA-II and ϵ-ranking varying ϵ. Similarly, **Figs. 8** and **9** show results for $K = 7$ and various M. From Fig. 6, we can see that $\mathcal{H}$ improves as we increase ϵ, eventually reaching maximum $\mathcal{H}^*$ at ϵ^*. Increasing ϵ above ϵ^* has the effect to gradually reduce $\mathcal{H}$, specially on $1 \leq K \leq 25$ landscapes (skewed bell-like curves with larger tail for $\epsilon > \epsilon^*$). However, $\epsilon > \epsilon^*$ on $K = \{0, 35, 50\}$ seems not to affect $\mathcal{H}$ (ramp-like curves). Thus, the region where ϵ^* lies in parameter space could be narrow or broad and depends on K. Note that for most bell-like curves, ϵ^* for different K (or an ϵ that leads to high $\mathcal{H}$) are clustered at close range. For example, ϵ^* for most K are in the range $[2.5, 4.5]$ on $M = 6$ objectives. Looking at Figs. 8 and 9, note that ϵ^* also depends on M.

Looking at $\mathcal{C}$ plots in Fig. 7, it is interesting to notice that most $\mathcal{C}(E, N)$ curves are bell-like. This suggests that, although there could be a broad range of values



(a) $M = \{3, 4, 5, 6\}$, $K = 7$ (b) $M = \{7, 8, 9, 10\}$, $K = 7$

Fig. 9 $\mathcal{C}$ between NSGA-II and ϵ-ranking varying ϵ, $M = \{3, 4, 5, 6, 7, 8, 9, 10\}$, $K = 7$.

for parameter ϵ that leads to high $\mathcal{H}$, the choice of ϵ would allow to focus on higher convergence or higher diversity compared to NSGA-II. Note that $\mathcal{C}(N, E)$ are close to 0.0 for most K (specially for large M). So, the choice of ϵ would not worsen convergence compared to NSGA-II (an exception could be $M = 2$, $K \leq 1$ and $M = 3$, $K = 0$ if ϵ is too large). Also, it is important to note that the value of ϵ that maximizes $\mathcal{H}$ is similar to the value of ϵ that maximizes $\mathcal{C}(E, N)$ for most K (and M). **Figure 10** shows the number of solutions $|\mathcal{F}_1^\epsilon|$ in the first front after applying ϵ-ranking for some values of ϵ on $M = 6$ objectives and $K = 10$ epistatic patterns per bit. The horizontal dashed line at 100 indicates the size of the parent population $|\mathcal{P}|$. From this figure, see that by increasing ϵ we can progressively reduce $|\mathcal{F}_1^\epsilon|$, i.e., the number of individuals assigned highest rank. It is interesting to notice that the steep performance improvement observed in Figs. 6 and 7 (see left side of the skewed bell-like curves for $K = 10$) correspond to values of ϵ that induce a better approximation of $|\mathcal{F}_1^\epsilon|$ to $|\mathcal{P}|$. See that, for $M = 6$ and $K = 10$, setting ϵ in the range $[0.03, 0.04]$ leads to a $|\mathcal{F}_1^\epsilon|$ that is close to the population size $|\mathcal{P}|$. Also, in Figs. 6 and 7 see that $\mathcal{H}$ and $\mathcal{C}$ are maximized in the same range of ϵ. A similar trend is observed for other values of K and M, suggesting that a good setting for ϵ is a value that induces $|\mathcal{F}_1^\epsilon|$ to be close to $|\mathcal{P}|$.

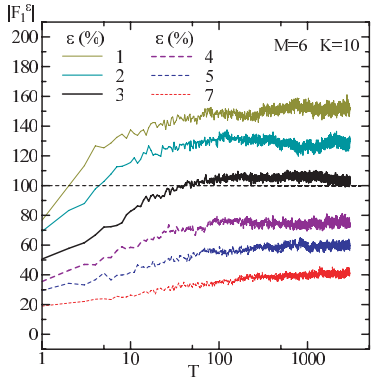


Fig. 10 Number of individuals $|\mathcal{F}_1^\epsilon|$ (over the generations) in the first front after ϵ-ranking, for several values of ϵ. $M = 6$, $K = 10$.

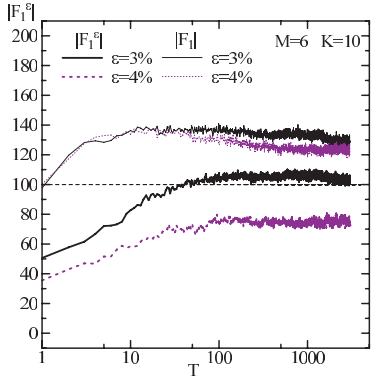


Fig. 11 Number of individuals $|\mathcal{F}_1|$ and $|\mathcal{F}_1^\epsilon|$ (over the generations) in the first front before and after ϵ-ranking, for values of ϵ in the range that maximize $\mathcal{H}$ and $\mathcal{C}$. $M = 6$, $K = 10$.

Figure 11 shows the number of individuals $|\mathcal{F}_1|$ and $|\mathcal{F}_1^\epsilon|$ that belong to the first front before and after ϵ-ranking, respectively, for values of ϵ in the range $[0.03, 0.04]$ where $\mathcal{H}$ and $\mathcal{C}$ are maximized on problems with $M = 6$ objectives and $K = 10$ epistatic interactions. Note that the number of individuals with highest rank $|\mathcal{F}_1|$ by conventional Pareto dominance is higher than $|\mathcal{P}|$ since the start of the run. Also, see that fixed values of ϵ induce larger differences

between $|\mathcal{F}_1|$ and $|\mathcal{F}_1^{\epsilon}|$ during the initial stages of the run. This is because the instantaneous population will cover wider areas of objective space as evolution proceeds.

The patterns we observe here are useful to understand better the optimization of problems with *many* objectives and for setting ϵ with an appropriate fixed value. However, it would be interesting in the future to study the effects of dynamically adjusting ϵ and to investigate adaptive methods to control it.

5.3 Absolute Values of $\mathcal{H}$ and Performance Scalability on K

In previous sections we focused on the ratio $\frac{\mathcal{H}(E)}{\mathcal{H}(N)}$ to discuss the relative gains on performance produced by the improved selection mechanism. In this section we analyze the absolute values of $\mathcal{H}$ to discuss the performance scalability of the algorithms as we increase the number of epistatic interactions. **Figure 12** illustrates the absolute values of $\mathcal{H}(N)$ and $\mathcal{H}(E)$ (set with $\epsilon = \epsilon^*$) over the number of epistatic interactions K on $M = 6$ objectives problems.

To understand these results, we first recall an important property of MNK-landscapes. Analysis by enumeration on $N = 20$ bits MNK-landscapes have shown that the trend of the hypervolume of the true Pareto front is to rapidly increase with K, from $K = 0$ to small values of K, remaining high for medium and large K [3),5)]. The hypervolume trend is expected to be similar on landscapes with a larger number of bits. This is a property directly related to the maximum

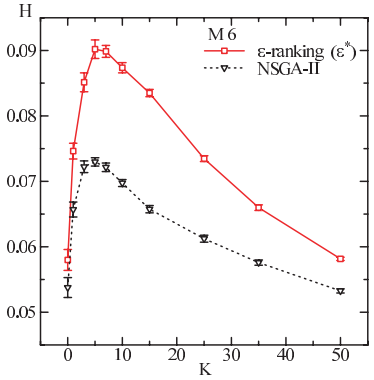


Fig. 12 Absolute values of $\mathcal{H}(N)$ and $\mathcal{H}(E)$ by NSGA-II and ϵ-ranking set with ϵ^* that achieves maximum $\mathcal{H}$, respectively.

achievable optima of single objective NK-landscapes [13)]. The achievable optima in NK-landscapes is lowest for $K = 0$ epistatic interactions (around 0.66 given by the central limit theorem). Increasing K to small values gradually bends the single objective landscapes and the achievable optima gets higher. Further increasing K to medium and large values bends no more the landscapes (though the number of local optima increase) and the achievable optima remains high.

From Fig. 12, see that hypervolume increases for $K \leq 5$, which is in accordance with the expected trend of the hypervolume. However, for $K \geq 7$ see that the value of the hypervolume decreases monotonically. These decreasing values are against the expected trend of the hypervolume and indicate that the search performance of the algorithms is worsening significantly as K increases. By including ϵ-ranking we have been able to improve significantly the performance of the evolutionary optimizer on $M \geq 3$ objectives for all $0 \leq K \leq 50$. However, Fig. 12 shows us that there is still work to be done to improve the performance of the algorithm on landscapes with medium and large K.

6. Comparison with Other Approaches

Recently, few methods have been proposed for many-objective optimization [7)–9)]. In the following we compare the performance of ϵ-ranking with two other methods that enhance selection in NSGA-II, namely Subvector Dominance Assignment (SVDOM) and Epsilon Dominance Assignment (EPSDOM) [8)]. Since ϵ-ranking also enhances selection and it is implemented in this work using the framework of NSGA-II, the comparison among ϵ-ranking, SVDOM, and EPSDOM would allow a clear understanding on the effects on performance of different strategies that enhance selection while keeping intact other components of the evolutionary algorithm.

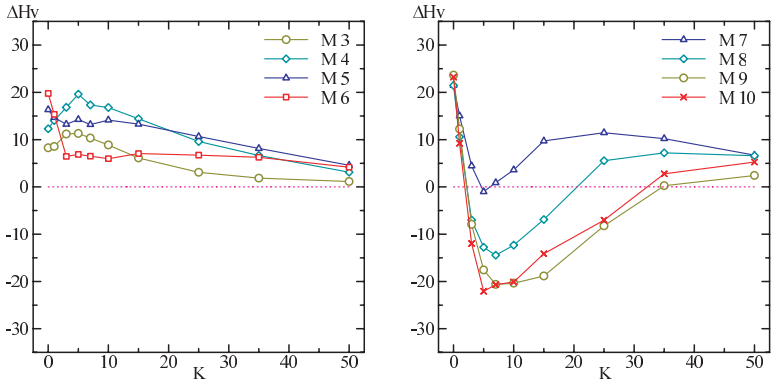
Before we analyze results by these algorithms, let us briefly highlight their main differences. NSGA-II uses two procedures to rank solutions [10)]. A primary ranking in NSGA-II is given by non-domination sorting and a secondary ranking by crowding distance of solutions. The secondary ranking is used when selection cannot discriminate based on the primary ranking of solutions. ϵ-ranking focuses on the primary ranking procedure, i.e., re-ranking of solutions after they have been initially ranked with non-domination sorting, and keeps crowding distance

as the secondary ranking procedure. On the other hand, SVDOM and EPSDOM modify NSGA-II's secondary ranking of solutions by using a substitute distance instead of crowding distance and keep the initial primary ranking given by non-domination sorting.

The substitute distance that determines the secondary ranking of solutions in SVDOM and EPSDOM are based on measurement procedures that calculate the highest degree to which a solution is nearly Pareto dominated by any other solution. SVDOM compares a solution $\boldsymbol{x}$ with other non-dominated solution $\boldsymbol{y}$, counting the number of objectives of $\boldsymbol{y}$ that are better than the corresponding objectives of $\boldsymbol{x}$, i.e., $c_y = count\{f_m(\boldsymbol{y}) > f_m(\boldsymbol{x}),\ m = 1, 2, \cdots, M\}$. Note that we maximize all objectives. The secondary ranking d_x^S of $\boldsymbol{x}$ is given by $d_x^S = \max\{c_y\}$ after comparing $\boldsymbol{x}$ with all other non-dominated solutions $\boldsymbol{y}$. A smaller d_x^S means a better rank. On the other hand, EPSDOM takes into account the magnitude of all objectives of $\boldsymbol{y}$ that are worse than the corresponding objectives of $\boldsymbol{x}$. That is, the magnitude that solution $\boldsymbol{y}$ needs to improve in order to dominate $\boldsymbol{x}$ is given by $e_y = \max\{f_m(\boldsymbol{y}) - f_m(\boldsymbol{x}) \mid f_m(\boldsymbol{y}) < f_m(\boldsymbol{x}),\ m = 1, 2, \cdots, M\}$. Now, the secondary ranking d_x^P of $\boldsymbol{x}$ is given by $d_x^P = \min\{e_y\}$ after comparing $\boldsymbol{x}$ with all other non-dominated solutions $\boldsymbol{y}$. A larger d_x^P means a better rank. The reader is referred to Ref. 8) for further details on SVDOM and EPSDOM.

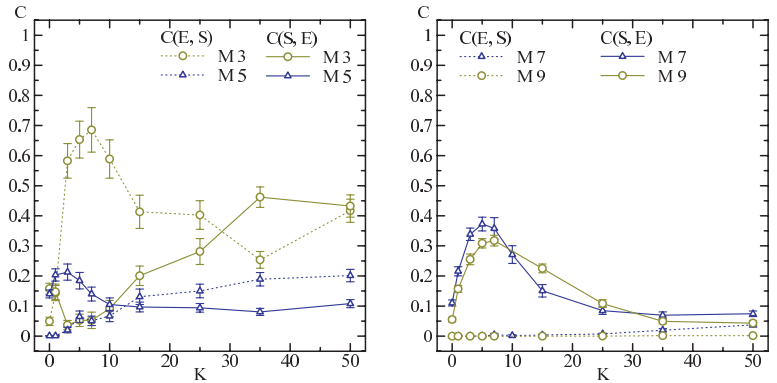
Figure 13 shows for $M = \{3, 4, 5, 6, 7, 8, 9, 10\}$ objectives the differential hypervolume $\Delta\mathcal{H}$ between ϵ-ranking (E) set with ϵ^* and SVDOM (S) normalized with respect to NSGA-II's hypervolume, $\Delta\mathcal{H} = \frac{\mathcal{H}(E)-\mathcal{H}(S)}{\mathcal{H}(N)} \times 100$. Note that $\Delta\mathcal{H} > 0$ indicates better hypervolume by ϵ-ranking, whereas a $\Delta\mathcal{H} < 0$ indicates better hypervolume by SVDOM. **Figure 14** shows the $\mathcal{C}$ values between ϵ-ranking (E) set with ϵ^* and SVDOM (S). For the sake of clarity, $\mathcal{C}$ values are included only for $M = \{3, 5, 7, 9\}$ objectives. Likewise, **Figs. 15** and **16** show results for ϵ-ranking (E) and EPSDOM (P).

From Fig. 13, note that ϵ-ranking achieves overall better hypervolume than SVDOM for $M \leq 7$, whereas for $8 \leq M \leq 10$ SVDOM achieves overall better hypervolume than ϵ-ranking (except for $K \geq 35$). Looking at Fig. 14, note that ϵ-ranking achieves better $\mathcal{C}$ values on $M \leq 5$ for some values of K. However, better $\mathcal{C}$ values are achieved by SVDOM for $M \geq 6$. Looking at Figs. 15 and 16, we can see that EPSDOM overall achieves better hypervolume and $\mathcal{C}$ values



(a) $M = \{3, 4, 5, 6\}$ (b) $M = \{7, 8, 9, 10\}$

Fig. 13 Differential hypervolume between ϵ-ranking (E) set with ϵ^* and SVDOM (S) normalized to NSGA-II's hypervolume.

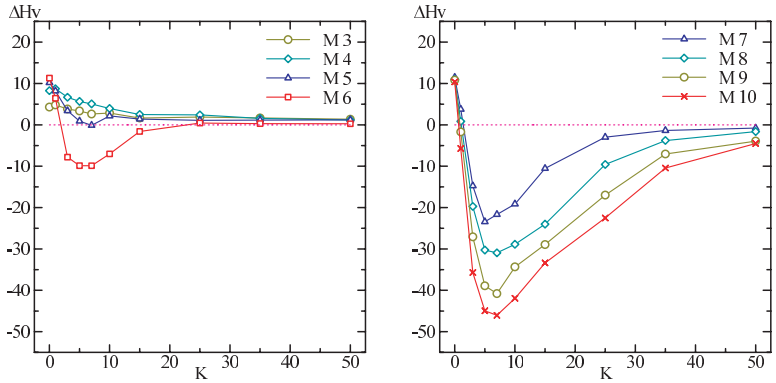


(a) $M = \{3, 5\}$ (b) $M = \{7, 9\}$

Fig. 14 $\mathcal{C}$ between ϵ-ranking (E) set with ϵ^* and SVDOM (S).

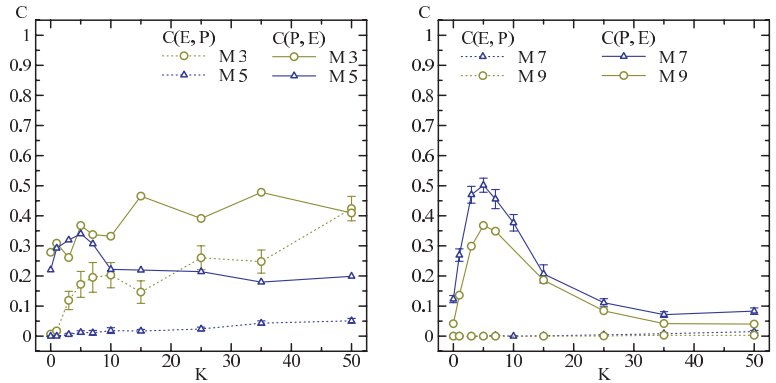
than SVDOM. Comparing ϵ-ranking and EPSDOM, note that ϵ-ranking achieves similar or better hypervolume on $M \leq 5$ objectives, whereas EPSDOM obtains better hypervolume on $M \geq 6$. Also note that EPSDOM achieves better $\mathcal{C}$ even on small M, where ϵ-ranking achieves better hypervolume.

From these results, at first glance it could seem that SVDOM and EPSDOM



(a) $M = \{3, 4, 5, 6\}$ (b) $M = \{7, 8, 9, 10\}$

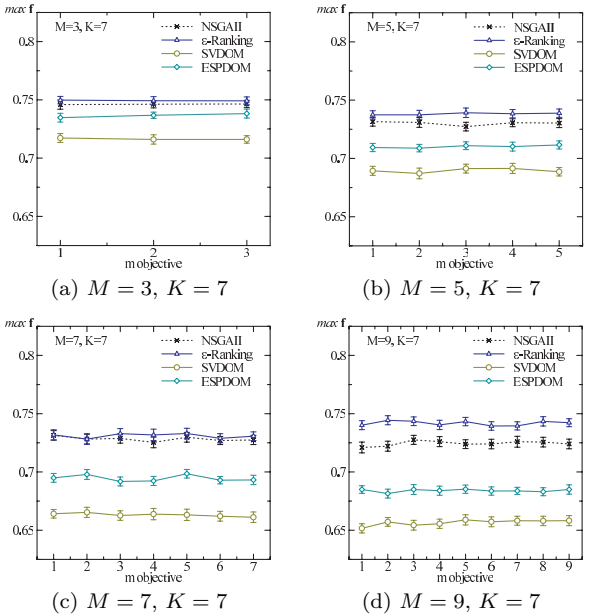
Fig. 15 Differential hypervolume between ϵ-ranking (E) set with ϵ^* and EPSDOM (P) normalized to NSGA-II's hypervolume.



(a) $M = \{3, 5\}$ (b) $M = \{7, 9\}$

Fig. 16 $\mathcal{C}$ between ϵ-ranking (E) set with ϵ^* and EPSDOM (P).

are a better optimizer than ϵ-ranking for larger M. However, the good values of hypervolume by SVDOM and EPSDOM are exclusively due to a better convergence of some solutions in a narrow region of objective space at the expense of diversity. To illustrate this important point, **Fig. 17** plots the maximum attained fitness values in each objective function by conventional NSGA-II, ϵ-ranking, SVDOM, and EPSDOM for $M = \{3, 5, 7, 9\}$ objectives and $K = 7$ epistatic interactions. Note that SVDOM and EPSDOM attain the highest values of $\mathcal{H}$ and $\mathcal{C}$ at $K = 7$ for most M. From Fig. 17, we can see that conventional NSGA-II achieves in all objectives better maximum-fitness values than SVDOM and EPSDOM. Note also that the gap between NSGA-II and SVDOM (or EPSDOM) increases with the number of objectives M of the problem. These results suggest that solutions found by SVDOM and EPSDOM tend to concentrate within an increasingly reduced space of objective space as the number of objectives of the problem increase. On the other hand, note that ϵ-ranking obtains similar or better maximum-fitness than conventional NSGA-II, with an increased gap in favor of ϵ-ranking as the number of objectives of the problem increase.



(a) $M = 3$, $K = 7$ (b) $M = 5$, $K = 7$

(c) $M = 7$, $K = 7$ (d) $M = 9$, $K = 7$

Fig. 17 Max. fitness in each objective by NSGA-II, ϵ-ranking, SVDOM, and EPSDOM. $M = \{3, 5, 7, 9\}$, $K = 7$.

Summarizing, SVDOM and EPSDOM especially are good methods to focus the search in an increasingly narrower part of the objective space as the number of

objectives of the problem increase. As a consequence, these methods can improve convergence of solutions, but at the expense of diversity. In fact, diversity by these methods is even worse than conventional NSGA-II for $M \geq 3$. On the other hand, ϵ-ranking can achieve better convergence and better diversity than NSGA-II for $M \geq 3$. Comparing SVDOM and EPSDOM, EPSDOM performs better both on convergence and diversity. Between ϵ-dominance and EPSDOM, there is not a clear winner. On the one hand, EPSDOM can find solutions with better convergence than ϵ-ranking within a narrower region of objective space. On the other hand, ϵ-ranking can find non-dominated solutions that EPSDOM cannot in a broader region of objective space. In both methods there is a trade-off on convergence and diversity, and their use should be decided depending on the kind of solutions we want the optimizer to find.

7. Computational Cost

In the previous sections we have compared the performance in terms of solution quality of NSGA-II, ϵ-ranking, SVDOM, and EPSDOM. In the following, we compare their computational cost. As mentioned above, ϵ-ranking, SVDOM, and EPSDOM are implemented within the framework of NSGA-II to enhance its selection procedure. From a computational stand point, the difference between NSGA-II and ϵ-ranking is the additional re-ranking procedure in the latter. Thus, a computational cost higher than NSGA-II's is expected in ϵ-ranking. However, it can be shown that the computational order of the additional re-ranking procedure is of order $\ll \mathcal{O}(MN^2)$. Note in **Procedure 1** that at each iteration of the loop in ϵ-sampling the number of comparisons is performed in an increasingly smaller set of non-dominated solutions, because sampled solutions as well as ϵ-dominated solutions are removed from the initial set of non-dominated solutions $\mathcal{A}$ and assigned to the sets $\mathcal{S}$ and $\mathcal{D}^{\epsilon}$, respectively.

SVDOM and EPSDOM modify NSGA-II's secondary ranking of solutions by using a substitute distance instead of crowding distance. In other words, any difference in computational cost between NSGA-II and SVDOM/EPSDOM is given by the differential computational order between crowding distance and the corresponding substitute distance. The computational order of applying crowding distance [10] to a set of non-dominated solutions is $\mathcal{O}(MN + MN \log N)$,

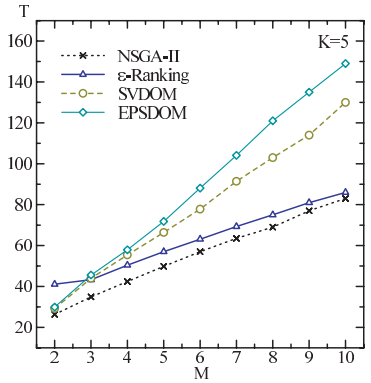


Fig. 18 Average run time in seconds by NSGA-II, ϵ-ranking, SVDOM, and EPSDOM. $M = \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and $K = 5$.

where N is number of solutions and M is the number of objectives. On the other hand, the computational order of the substitute distances in SVDOM and EPSDOM [8] are $\mathcal{O}(MN^2)$. Since $MN^2 > MN + MN \log N$, SVDOM and EPSDOM have an computational cost higher than NSGA-II of order $\mathcal{O}(MN^2 - MN - MN \log N) = \mathcal{O}(MN(N - 1 - \log N)) \sim \mathcal{O}(MN^2)$ for large N.

Figure 18 shows the average number of seconds per run by NSGA-II, ϵ-ranking, SVDOM, and EPSDOM for $M = \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and $K = 5$ epistatic interactions. From this figure, note that NSGA-II requires lower computational time among all algorithms. Also see that the computational time of the algorithms increases with the number of objectives of the problem, which is expected for algorithms that use Pareto dominance. Between ϵ-ranking and SVDOM/EPSDOM, note that overall ϵ-ranking is faster than SVDOM and EPSDOM.

8. Conclusions

In this work, we have proposed a method to fine grain the ranking of solutions in Pareto dominance MOEAs aiming to improve their performance on *many* objectives problems. The re-ranking method uses a randomized sampling procedure to increase selection probabilities of some of the solutions, while trying to keep a uniform search effort towards the different zones of objective space represented in

the instantaneous population. We enhanced NSGA-II with the proposed method and tested its performance on MNK-Landscapes with $2 \leq M \leq 10$ objectives, $N = 100$ bits and $0 \leq K \leq 50$ epistatic interactions. We showed that both convergence and diversity of the obtained solutions can improve remarkably on problems with $3 \leq M \leq 10$ objectives for any level of epistatic interactions.

We also compared results by the proposed method with Subvector Dominance Assignment (SVDOM) and Epsilon Dominance Assignment (EPSDOM), showing that SVDOM and EPSDOM can improve convergence of some solutions but at the expense of diversity, whereas ϵ-ranking can find solutions in a broader region of objective space but at the expense of convergence of some solutions. Furthermore, we discussed the computational time of the algorithms, showing that ϵ-ranking is overall faster than SVDOM and EPSDOM.

As future works, we would like to pursue adaptive methods to control the expanded dominance region of the sampled solutions and investigate ways to further improve convergence, especially on $M \geq 7$ objective problems. Also, we would like to compare the proposed method with other approaches that are being developed for *many* objective optimization.

Acknowledgments This study was performed through Special Coordination Funds for Promoting Science and Technology of the Ministry of Education, Culture, Sports, Science and Technology, of the Japanese Government.

References

1) Deb, K.: *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Chichester, West Sussex, England (2001).
2) Coello, C., Van Veldhuizen, D. and Lamont, G.: *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, Boston (2002).
3) Aguirre, H. and Tanaka, K.: Insights on Properties of Multiobjective MNK-Landscapes, *Proc. 2004 IEEE Congress on Evolutionary Computation*, IEEE Service Center, pp.196–203 (2004).
4) Aguirre, H. and Tanaka, K.: Selection, Drift, Recombination, and Mutation in Multiobjective Evolutionary Algorithms on Scalable MNK-Landscapes, *Proc. 3rd Int'l Conf. on Evolutionary Multi-Criterion Optimization*, LNCS, Vol.3410, pp.355-369, Springer (2005).
5) Aguirre, H. and Tanaka, K.: Working Principles, Behavior, and Performance of MOEAs on MNK-Landscapes, *European Journal of Operational Research*, Vol.181, No.3, pp.1670–1690, Elsevier (Sep. 2007).
6) Purshouse, R.C. and Fleming, P.J.: Conflict, Harmony, and Independence: Relationships in Evolutionary Multi-criterion Optimisation, *2nd Int'l Conf. on Evolutionary Multi-Criterion Optimization*, LNCS, Vol.2632, pp.16–30, Springer (Apr. 2003).
7) Hughes, E.J.: Evolutionary Many-Objective Optimisation: Many Once or One Many?, *Proc. 2005 IEEE Congress on Evolutionary Computation*, Vol.1, pp.222–227, IEEE Service Center (Sep. 2005).
8) Koppen, M. and Yoshida, K.: Substitute Distance Assignments in NSGA-II for Handling Many-Objective Optimization Problems, *Proc. 4th Int'l Conf. on Evolutionary Multi-Criterion Optimization*, LNCS, Vol.4403, pp.727–741, Springer (2007).
9) Wagner, T., Beume, N. and Naujoks, B.: Pareto-, Aggregation, and Indicator-Based Methods in Many-Objective Optimization, *Proc. 4th Int'l Conf. on Evolutionary Multi-Criterion Optimization*, LNCS, Vol.4403, pp.742–756, Springer (2007).
10) Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II, *KanGAL report 200001* (2000).
11) Yu, P.L.: Cone Convexity, Cone Extreme Points, and Nondominated Solutions in Decision Problems with Multiobjectives, *Journal of Optimization Theory and Applications*, Vol.14, No.3, pp.319–377 (1974).
12) Laumanns, M., Thiele, L., Deb, K. and Zitzler, E.: Combining Convergence and Diversity in Evolutionary Multi-objective Optimization, *Evolutionary Computation*, Vol.10, No.3, pp.263–282 (Fall 2002).
13) Kauffman, S.A.: *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press (1993).
14) Deb, K., Thiele, L., Laumanns, M. and Zitzler, E.: Scalable Multi-Objective Optimization Test Problems, *Proc. 2002 Congress on Evolutionary Computation*, pp.825–830, IEEE Service Center (2002).
15) Zitzler, E.: *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, Ph.D. thesis, Swiss Federal Institute of Technology, Zurich (1999).
16) Fonseca, C., Paquete, L. and López-Ibáñez, M.: An Improved Dimension-sweep Algorithm for the Hypervolume Indicator, *Proc. 2006 IEEE Congress on Evolutionary Computation*, pp.1157–1163, IEEE Service Center (2006).

(Received November 19, 2008)
(Revised January 1, 2009)
(Accepted February 6, 2009)

Hernán Aguirre received the Engineer degree in computer systems from Escuela Politécnica Nacional, Quito, Ecuador in 1992 and the M.S. and Ph.D. degrees from Shinshu University, Japan, in 2000 and 2003, respectively. Currently, he is an assistant professor at Shinshu University. His research interests include evolutionary computation, multidisciplinary design optimization, computational intelligence, information security, parallel computing, and their applications. He is a member of IEEE, ACM-SIGEVO, and IPSJ.

Kiyoshi Tanaka received his B.S. and M.S. degrees in Electrical Engineering and Operations Research from National Defense Academy, Yokosuka, Japan, in 1984 and 1989, respectively. In 1992, he received the D.E. degree from Keio University, Tokyo, Japan. In 1995, he joined the Department of Electrical and Electronic Engineering, Faculty of Engineering, Shinshu University, Nagano, Japan, where he is currently a professor. His research interests include image and video processing, information hiding, evolutionary computation, chaos & fractals, and their applications. He is a member of IEEE, IEICE, IPSJ and IIEEJ. He is the vice-chairman of journal editing committee of IIEEJ.