Working principles, behavior, and performance of MOEAs on MNK-landscapes

Hernán E. Aguirre *, Kiyoshi Tanaka

Faculty of Engineering, Shinshu University, 4-17-1 Wakasato, Nagano 380-8553, Japan

Available online 18 September 2006

Abstract

This work studies the working principles, behavior, and performance of multiobjective evolutionary algorithms (MOEAs) on multiobjective epistatic fitness functions with discrete binary search spaces by using MNK-landscapes. First, we analyze the structure and some of the properties of MNK-landscapes under a multiobjective perspective by using enumeration on small landscapes. Then, we focus on the performance and behavior of MOEAs on large landscapes. We organize our study around selection, drift, mutation, and recombination, the four major and intertwined processes that drive adaptive evolution over fitness landscapes. This work clearly shows pros and cons of the main features of MOEAs, gives a valuable guide for the practitioner on how to set up his/her algorithm, enhance MOEAs, and presents useful insights on how to design more robust and efficient MOEAs.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Evolutionary computations; Multiobjective evolutionary algorithms; Multiobjective combinatorial optimization; MNK-landscapes; Epistasis; Non-linear multiobjective fitness functions; Discrete binary search spaces; Selection; Drift; Mutation; Recombination

1. Introduction

Epistasis in the context of evolutionary algorithms (EAs) describes non-linearities in fitness functions due to changes in the values of interacting bits. Epistasis is recognized as an important factor that makes a problem difficult for optimization algorithms and its influence on the performance of single objective EAs is being increasingly investigated (Davidor, 1991; Manderick et al., 1991; Altenberg, 1994; De Jong et al., 1997; Altenberg, 1997; Merz and Freisleben, 1998; Heckendorn et al., 1999; Smith and Smith, 1999; Mathias et al., 2001; Aguirre and Tanaka, 2003). Particularly, Kauffman’s NK-landscapes model of epistatic interactions (Kauffman, 1993) has been the center of several studies, both for the statistical properties of the generated landscapes and for their EA-hardness. Studies on the behavior of single objective EAs on NK-landscapes have proved useful to advance our understanding of EA’s working principles and served to design robust and better algorithms (see Section 2.1).

Contrary to the single objective case, studies concerning epistasis within the context of multiobjective optimization are quite few and its effects are still far from being well understood. Due to the characteristics and requirements particular to multiobjective
optimization (Knowles and Corne, 2002; Zitzler et al., 2003; Bosman and Thierens, 2003), our knowledge of the effects of epistasis on landscape’s properties and its influence on the performance of single objective optimizers is not readily transferable to the multiobjective domain. Hence, there is the need to comprehend better the implications of epistasis on the properties of multiobjective landscapes as well as its influence on the working mechanism, performance, and behavior of multiobjective optimizers. Efforts in this direction include works on deception problems using multiobjective Bayesian optimization algorithms (Khan et al., 2002; Khan, 2003) and multiobjective messy genetic algorithms (Day et al., 2004; Day and Lamont, 2005). Also, recently Kauffman’s NK-landscapes model of epistatic interactions has been extended to multiobjective MNK-landscapes (Aguirre and Tanaka, 2004a) and the behavior of multiobjective random one-bit climbers (moRBCs) has been studied on these landscapes (Aguirre and Tanaka, 2004b) (see Section 2.2).

This work studies the working principles, behavior, and performance of multiobjective evolutionary algorithms (MOEAs) on epistatic multiobjective fitness functions with discrete binary search spaces by using MNK-landscapes. First, in order to understand and explain better the behavior of the algorithms, we analyze the structure and some of the properties of MNK-landscapes under a multiobjective perspective by using enumeration on small landscapes. Then, we focus on the working mechanisms, performance, and behavior of MOEAs on large landscapes. We organize our study around selection, drift, mutation, and recombination; the four major and intertwined processes that drive adaptive evolution over fitness landscapes (Kauffman, 1993). In this work we consider MOEAs in which selection incorporates elitism and it is biased by Pareto dominance and a diversity preserving strategy in objective space (Deb, 2001; Coello et al., 2002). Recombination is implemented as two-point crossover and mutation as the standard bit flipping method and/or as a local search-like strategy. In addition to these features that are explicit to the algorithm design, drift is also considered in our study because it is an important process implicit to all stochastic algorithms working on finite small populations.

The main goals of this paper are as follows. Verify the importance of elitism in multiobjective optimization clarifying some of its undesired side effects. Analyze front level and objective space diversity criteria to bias selection, contrasting with the properties of the landscapes. Understand the contribution to performance of distributing selection pressure more fairly among individuals by eliminating duplicate individuals. Finally, verify the contribution of crossover, short-range mutation, and local search-like mutation. This work clearly shows pros and cons of elitist Pareto-based MOEAs, gives a valuable guide for the practitioner on how to set up his/her algorithm, enhance MOEAs, and presents useful insights on how to design more robust and efficient MOEAs. Parts of this work have been presented in Aguirre and Tanaka (2004a,b) and Aguirre and Tanaka (2005).

2. Multiobjective MNK-landscapes

2.1. Epistasis and single objective NK-landscapes

Epistasis is a term used in biology to describe a range of non-additive phenomena due to the non-linear inter-dependence of gene values, i.e. the expression of one gene masks the genotypic effect of another. In the context of EAs this terminology is used to describe non-linearities in fitness functions due to changes in the values of interacting bits. An implication of epistatic interactions among bits is that the fitness function develops conflicting constraints (Altenberg, 1997). That is, a mutation in one bit may improve its contribution to fitness but may decrease the contributions of other bits with which it interacts. Hence, epistatic interactions increase the difficulty in trying to optimize all bits simultaneously.

Kauffman (1993) designed a family of problems, the NK-landscapes, to explore how epistasis is linked to the ruggedness of search spaces. Here epistasis corresponds to the degree of interaction between genes (bits), and ruggedness is related to local optima, their number, and their density. More formally, a NK-landscape is a function $f : \mathbb{B}^N \to \mathbb{R}$ where $\mathbb{B} = \{0, 1\}$, $N$ is the bit string length, and $K$ is the number of bits in the string that epistatically interact with each bit. In NK-landscapes the fitness of the bit string $f(x)$ is the average fitness of all its bits and is given by the following equation:

$$f(x) = \frac{1}{N} \sum_{j=1}^{N} f_j(x_j, z_1^{(j)}, z_2^{(j)}, \ldots, z_K^{(j)}),$$

where $f_j : \mathbb{B}^{K+1} \to \mathbb{R}$ gives the fitness contribution of bit $x_j$, and $z_1^{(j)}, z_2^{(j)}, \ldots, z_K^{(j)}$ are the $K$ bits interact-
ing with bit $x_j$ in the string $x$. Note that the fitness of each bit of the string depends on the combination of its on value and the values of the interacting bits. NK-landscapes are stochastically generated and usually the fitness contribution $f_j$ of bit $x_j$ is a number between $[0.0, 1.0]$ drawn from a uniform distribution.

After several years of research, the properties of NK-landscapes have become well known and within the EA community this model of epistatic interactions is considered as a good test problem generator for single objective combinatorial optimization problems. With NK-landscapes, we can evaluate the overall performance of algorithms on sub-classes of epistatic problems observing their behavior as we increase the non-linearity (ruggedness and multimodality) of the fitness landscapes. Moreover, epistasis is also related to the underlying representation of individuals and its understanding is fundamental to the design of appropriate representations (Altenberg, 1994). Comparing EAs and testing their main working principles on sub-classes of problems with well known properties, such NK-landscapes, have permitted to draw more meaningful conclusions than testing the algorithms in few problems and have paved ways to develop new and improved EAs (De Jong et al., 1997; Merz and Freisleben, 1998; Heckendorn et al., 1999; Smith and Smith, 1999; Mathias et al., 2001; Aguirre and Tanaka, 2003).

It should be mentioned that there are robust EAs for single objective epistatic NK-landscapes. In Aguirre and Tanaka (2003), it was shown on landscapes with $N = 96$ bits that postponing drift combined with an appropriate selection pressure make EAs quite robust on epistatic problems. Even simple EAs including these two features were shown to perform better than random bit climbers for a broad range of classes of problems ($K \geq 4$). In addition, the concurrent interaction of crossover with varying mutation was shown to improve further the performance of EAs, especially in the range $12 < K < 32$. These latest results Aguirre and Tanaka, 2003 rectify initial misleading conclusions suggesting that EAs are doomed to perform badly, even worse than random bit climbers, on NK-landscapes (Heckendorn et al., 1999; Mathias et al., 2001).

### 2.2. Extension to multiobjective MNK-landscapes

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

$$\text{maximize } f(x) = (f_1(x), f_2(x), \ldots, f_M(x)), \quad (2)$$

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

A multiobjective MNK-landscape (Aguirre and Tanaka, 2004a) is a multiobjective combinatorial optimization problem with $M$ objectives, where each objective function is determined by a different instance of a NK-landscape over the same bit string. Specifically, a MNK-landscape is defined as a vector function mapping binary strings into $M$ real numbers $f(\cdot) = (f_1(\cdot), f_2(\cdot), \ldots, f_M(\cdot)) : \mathcal{B}^N \rightarrow \mathbb{R}^M$, 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. $K = \{K_1, \ldots, K_M\}$ is a set of integers where $K_i$ ($i = 1, 2, \ldots, M$) is the number of bits in the string that epistatically interact with each bit in the $i$th landscape. Each $f_i(\cdot)$ is expressed as the average of $N$ functions by

$$f_i(x) = \frac{1}{N} \sum_{j=1}^{N} f_{ij}(x_j, z_{i,1}^{(i)}, z_{i,2}^{(i)}, \ldots, z_{i,K_i}^{(i)}) \quad (3)$$

where $f_{ij} : \mathcal{B}^{K_i+1} \rightarrow \mathbb{R}$ gives the fitness contribution of bit $x_j$ to $f_i(\cdot)$, and $z_{i,1}^{(i)}, z_{i,2}^{(i)}, \ldots, z_{i,K_i}^{(i)}$ are the $K_i$ bits interacting with bit $x_j$ in the string $x$. The fitness contribution $f_{ij}$ 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 $x$ expressed by a Kauffman’s NK-landscape model of epistatic interactions, see (1). Fig. 1 shows an example of the fitness functions $f_{1,3}$ and $f_{2,3}$ associated to bit $x_3$ contributing to the first objective function $f_1(\cdot)$ and second one $f_2(\cdot)$, respectively, based on a different epistatic model for each objective.

From a multiobjective random test problem generator standpoint (Deb et al., 2002), 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 binary search spaces.

For a given $N$, we can tune the ruggedness of the fitness function $f(\cdot)$ 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 multipeaked landscapes. In 3 we give a detailed account on the structure and properties of MNK-landscapes, revealing how the difficulty of the problem and the Pareto front itself arise from the interaction among bits in different objectives.

Besides defining \( M, N \) and \( K_i \) for each \( f_i(\cdot) \), it is also possible to arrange the epistatic pattern between bit \( x_i \) and the \( K_i \) other interacting bits. That is, the distribution \( D_i = \{ \text{random, nearest neighbor} \} \) of \( K_i \) bits among \( N \). Thus, \( M, N, K = \{ K_1, K_2, \ldots, K_M \} \), and \( D = \{ D_1, D_2, \ldots, D_M \} \), completely specify a multiobjective MNK-landscape. 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 binary search spaces.

MNK-landscapes have been used to study the effects of elitism and population climbing. In Aguirre and Tanaka (2004b) it was shown that multiobjective random bit climbers (moRBCs) using a population for restarts performs better than moRBCs with minimum population \((1 + 1)\) supported by the archive as a reference memory. It was also shown that the population provides a non-dominated search front which coverage implicitly pushes the algorithm towards higher fronts and gives the random bit climber a more far-reaching capability to generate non-dominated solutions.

### 2.3. Multiobjective optimization concepts

Important concepts in multiobjective optimization are dominance, Pareto optimal solution, Pareto optimal set, and true Pareto front. Assuming without loss of generality that all objectives are maximized, these concepts can be defined as follows. Let us denote \( \mathcal{X} \subseteq \mathcal{P} \) a set of solutions and \( x, y \in \mathcal{X} \). Solution \( x \) dominates \( y \) if and only if the two following conditions are true:

1. \( \forall m \in \{1, \ldots, M\} \ f_m(x) \geq f_m(y) \),
2. \( \exists m \in \{1, \ldots, M\} \ f_m(x) > f_m(y) \).

In the following we denote \( x \) dominates \( y \) by \( f(x) \succeq f(y) \). A solution vector \( x^* \in \mathcal{X} \) is Pareto optimal if there is no solution that dominates it. More formally, \( x^* \in \mathcal{X} \) is Pareto optimal if \( \neg \exists y \in \mathcal{X} : f(y) \succeq f(x^*) \). The presence of multiple conflicting objectives gives rise to several Pareto optimal solutions. The Pareto optimal set of \( \mathcal{X} \) is defined by

\[
\mathcal{P}_1 = \{ x^* \in \mathcal{X} | \neg \exists y \in \mathcal{X} : f(y) \succeq f(x^*) \}.
\]

The Pareto front is the corresponding set in objective space of the Pareto optimal set \( \mathcal{P}_1 \) and it is defined by

\[
\mathcal{F}_1 = \{ f(x^*) = (f_1(x^*), f_2(x^*), \ldots, f_M(x^*)) | x^* \in \mathcal{P}_1 \}.
\]

In the case that the set of solutions \( \mathcal{X} \) refers to all solutions in the solution space \( \mathcal{P} \) of a problem, i.e. \( \mathcal{X} = \mathcal{P} \), we call \( \mathcal{F}_1 \) as the true Pareto front.

### 3. Structure and properties of MNK-landscapes by enumeration

In this section we analyze the structure and some properties of MNK-landscapes studying how the parameters of the landscapes relate to multiobjective concepts such as number of non-dominated fronts, number of non-dominated solutions, shape of the fronts, accessibility to the Pareto optimal front, correlation between fronts, correlations within fronts, and so on. In our methodology we use enumeration on landscapes with \( M = \{2, 3, 5\} \)
objectives and \( N = \{10, 15, 20\} \) bits, varying the number of epistatic interactions \( K_i = K \) from 0\% to 50\% of \( N \) simultaneously in all objectives \((K_1 = K_2 = \cdots = K_M = K)\). The epistatic pattern among bits is random for all objectives \((D_1 = D_2 = \cdots = D_M = \text{random})\). We analyze landscapes up to \( M = 5 \) objectives because we want to understand the landscapes and the performance of the algorithms as we move from multiple to many objectives optimization problems. Results presented in this section are averages over 30 different instances of a problem randomly generated for each combination of \( M, N \) and \( K \). Error bars indicate 95\% confidence intervals.

For each landscape we enumerate the search space \( S \) classifying solutions into non-dominated fronts. The first front \( F_1 \) is the true Pareto front and corresponds to the optimal Pareto set \( P_1 \) of \( S \). The subsequent fronts \( F_j, j > 1 \), contain lower level non-dominated solutions and are obtained by disregarding solutions corresponding to the previous higher level non-dominated fronts, i.e. \( F_j \) is the corresponding set of \( P_j \) in objective space, where \( P_j, j > 1 \), is the Pareto optimal set obtained from \( S = \bigcup_{j=1}^{\infty} P_j \).

First we look into the effect of varying the number of objectives \( M \) on the number of solutions per front and number of fronts. Fixing the size of the search space and increasing the number of objectives \( M \), we expect fewer but denser non-dominated fronts. An interesting and important question is the magnitude of the increase in number of solutions per front and the decrease in number of fronts. In order to have an insight on this, Fig. 2 illustrates the average number of solutions \(|P_j|\) in front \( F_j \) for landscapes with \( M = \{2, 3, 5\} \) objectives, \( N = 20 \) bits, and \( K = \{0, 3, 10\} \) epistatic interactions. From this figure, it can be seen that the number of solutions per front increases and the number of fronts decreases in one order of magnitude varying the number of objectives \( M \) from 2 to 3, and also in one order of magnitude from 3 to 5, for all values of epistatic interactions. For example, for \( K = 0 \) the average number of solutions in the true Pareto front \( F_1 \) are 32, 350, and 6341 for 2, 3, and 5 objectives, respectively. Likewise, for \( K = 0 \) the average number of fronts are 1936, 218, and 34. Similar trends are obtained for landscapes with \( N = 10 \) and \( N = 15 \) bits.

Second, we look into the effect of the number of epistatic interactions \( K \). From Fig. 2, we can see that for all values of \( M \) the number of solutions reduces with \( K \), especially in the top non-dominated fronts. This is better illustrated for the true Pareto front \( F_1 \) in Fig. 3, which shows the percentage ratio of number of solutions \(|P_1|\) in the true Pareto front \( F_1 \) to the total number of solutions in the search space \(|S| = 2^N\), for several values of \( K \). From Fig. 3 we also observe that the variance of the number of solutions is larger for small values of \( K \) than for high values of \( K \).

The effect of \( K \) on the number of solutions per front is explained from the kind of landscape yielded by the number of epistatic interactions. For \( K = 0 \), we have in each objective a single peaked, smooth, highly correlated fitness landscape (fitness of one-bit neighbor solutions are close to one another). Since we set the same \( K \) for all

![Fig. 2. Effect of \( M \). Average number of fronts and average number of solutions per front, \( M = \{2, 3, 5\}, N = 20 \).](image)

![Fig. 3. Effect of \( K \). Percentage of the average number of solutions \(|P_1|\) in the first front \( F_1 \), \( M = \{2, 3, 5\}, N = 20 \), \( K = 0 \), \( \ldots \), 50\% \( N \).](image)
objectives, we can think of similar landscapes shifted in genotype space. Because fitness would change smoothly in each objective, non-dominated solutions would be close to one another in genotype space forming broad regions of non-domination in the multiobjective landscape. Increasing $K$, the achievable optima, the number of peaks, and the ruggedness of the landscapes increase (Kauffman, 1993) for each objective $i$, narrowing the regions of non-domination in the multiobjective landscape. This is illustrated in Fig. 4, where the plots show the fitness of solutions in the first objective function $f_1$ over their Hamming distance to the optimum solution in function $f_1$. Results are presented for solutions in the three first fronts in landscapes with $M = \{2, 3\}$, $N = 20$, and $K = \{0, 3\}$. Note from the plots that for $K = 0$ fitness $f_1$ reduces as Hamming distance increases and that the solutions in the fronts seem contiguous in objective space and Hamming distance. On the other hand, for $K = 3$ (15%N), we can observe the presence of high fitness peaks far away from the optimum and that the solutions in the fronts become discontiguous in objective space and Hamming distance. Thus, $K$ also affects contiguity of non-dominated solutions in genotype and phenotype space.

To explain the high variance in number of solutions for small $K$, we look into the closeness on genotype space of the global optimum of each one of the objectives. Fig. 5 shows scatter plots of the number of solutions $|\mathcal{P}_1|$ in the true Pareto front $\mathcal{F}_1$ over the Hamming distance between optimum solutions in $f_1$ and $f_2$, for $M = 2$ and $K = \{0, 1, 3, 7\}$. There are 30 points in each plot, one per instance of a problem. These plots also include a first degree polynomial approximation of the data. Note that for small values of epistatic interactions there is a positive correlation between the distance of the highest fitness individual in each single objective landscape and the number of solutions in the

![Fig. 4. Hamming distance to the optimum solution on fitness function $f_1$. $M = \{2, 3\}$ $N = 20$, $K = \{0, 3\}$.](image-url)
true Pareto front. That is, the farther the distance between optimum solutions (in each objective) the higher the number of solutions \( |\mathcal{P}_1| \) in \( \mathcal{F}_1 \). In other words, the region of non-domination is broader for optimum solutions that lie far apart in genotype space. Increasing \( K \) this correlation reduces and eventually disappears because the number of peaks and their ruggedness increase with \( K \) in each objective, which narrow and disperse uniformly the non-domination regions between peaks of different objectives.

Another effect of \( K \) can be seen on the shape of the fronts. Fig. 6 shows the true Pareto front for one of the problems in landscapes with \( M = 2 \) objectives, \( N = 20 \) bits, and \( K = \{0, 1, 3, 7\} \) epistatic interactions. From this figure, it can be seen that \( K = 0 \) leads to convex Pareto fronts (and contiguous in objective space as explained above). Increasing \( K \), we can see that non-convex regions appear in the fronts (and discontiguities in objective space augment).

To better understand the effect of the number of epistatic interactions on the non-dominated fronts, we look at one-bit neighbor solutions. Let \( \mathcal{F}_j \) and \( \mathcal{F}_k \) be the \( j \)th and \( k \)th non-dominated fronts, respectively. \( \mathcal{P}_j^k \) is defined as the set of solutions in the \( j \)th front that are one-bit neighbors to at least one solution in the \( k \)th front. More formally,

\[
\mathcal{P}_j^k = \{ x \in \mathcal{P}_j \mapsto \exists y \in \mathcal{P}_k \mapsto h(x,y) = 1 \},
\]

where \( h() \) is the Hamming distance. The ratio \( z_j^k \) estimates a minimum degree of one-bit correlation between the \( j \)th and \( k \)th fronts and it is calculated by

\[
z_j^k = \frac{|\mathcal{P}_j^k|}{|\mathcal{P}_j|}.
\]
Fig. 7 plots the ratio $a_j^1$ illustrating the one-bit correlation from front $F_j$, $j \geq 2$, to the true Pareto front $F_1$. Results are presented for landscapes with $M = 3$, $N = 20$, and $K = \{0, 1, 3, 5, 7, 10\}$. From this figure it can be seen that $a_j^1$ is higher for the second front $F_2$ and reduces rapidly with front number $j$ for all values of epistatic interactions. For example, for $K = 0$ epistatic interactions, 98% of solutions in front $F_2$ are one-bit correlated to at least one solution in $F_1$ while only 6% of solutions in front $F_{10}$ are. It should be said that the 10 top fronts amount to 5% of the total number of non-dominated fronts. Also, it is interesting to note that $a_j^1$ reduces although the number of solutions in front $F_j$ increases with $j$, as shown in Fig. 2.

From Fig. 7, we can also see that the one-bit correlation to front $F_1$ reduces with $K$. For example, for $K = 0$, $K = 3$, and $K = 5$ the one-bit correlated solutions to front $F_1$ from $F_2$ are 98%, 72%, and 51%, respectively.

Fig. 8 plots the ratio $a_{j+1}^1$ illustrating the one-bit correlation between adjacent fronts and Fig. 9 plots
the ratio $\alpha$ showing the correlation within fronts. Both figures show results for landscapes with $M = \{2, 3\}$ objectives and $N = 20$ bits. For $M = 2$ we include the top 1000 fronts and for $M = 3$ the top 100 fronts, which amount to almost 50% of the total number of fronts in both cases. From these two figures it is worth noting the following. The correlation between contiguous fronts and the correlation within fronts decrease with the number of epistatic interactions. The correlations also reduce with front number $j$. The correlation within fronts ($\alpha_j$) is slightly higher than the correlation between fronts ($\alpha_{j+1}$) for all values of epistatic interactions. Also, note that increasing $M$ the fronts become more correlated.

A summary of the observed properties is as follows:

1. The number of solutions per front increases in one order of magnitude varying $M$ from 2 to 3, and also in one order of magnitude from 3 to 5. Conversely, and in similar proportions, the number of non-dominated fronts reduces with $M$.

2. The number of solutions reduces with $K$, especially in the true Pareto front and other high level non-dominated fronts.

3. $K$ is correlated to the variance in number of solutions per front. For small $K$, the variance is high and the number of solutions per front depends on the distance among optimum solutions in each objective. The variance reduces with $K$.

4. Other effects of $K$ can be seen on the contiguity of solutions in genotype and phenotype space. Optimal solutions and their true Pareto fronts become more discontinuous with $K$.

5. $K$ also affects the shape of the fronts. For small $K$ the fronts are convex. Increasing $K$, non-convex regions appear in the fronts.

Fig. 8. Average correlation between adjacent fronts (approximately 50% of the fronts are depicted). $N = 20$, $M = \{2, 3\}$ from left to right.

Fig. 9. Average correlation within fronts (approximately 50% of the fronts are depicted). $N = 20$, $M = \{2, 3\}$ from left to right.
(6) The number of solutions per front and the number of fronts increase in approximately one order of magnitude varying \( N \) from 10 to 20. However, the size of the search space \( 2^N \) increase in three orders of magnitude. Thus, the fraction of solutions in the true Pareto front to the total number of solution in the search space become smaller as \( N \) increases. The same is true for other high level non-dominated fronts.

(7) One-bit correlations to the true Pareto front from lower level non-dominated fronts reduce rapidly with the front’s level of non-dominance. These correlations also reduces with \( K \).

(8) One-bit correlations within fronts are slightly higher than one-bit correlations between contiguous fronts. Both kinds of correlation reduce with \( K \) and increase with \( M \).

The structure and properties highlighted in this section will be useful to understand and explain better the behavior and performance of MOEAs on MNK-landscapes with larger \( N \) presented in the following sections.

4. The Algorithms

In order to analyze selection, drift, recombination, and mutation in elitist Pareto-based MOEAs and obtain a broad perspective on their performance on multiobjective epistatic landscapes, we choose NSGA-II (Deb, 2000) and SPEA2 (Zitzler et al., 2001) as two representative algorithms. In addition, for comparison we also present results by moRBC(\( \gamma; 1 + 1 \)) (Aguirre and Tanaka, 2004b), a multiobjective random one-bit climber that uses a population for restarts. In the following we include a brief description of the algorithms.

4.1. NSGA-II

NSGA-II keeps at the \( t \)th generation a parent population \( \mathcal{F}(t) \) and an offspring population \( \mathcal{R}(t) \), both of same size \( \mu \). The parent population \( \mathcal{F}(t + 1) \) at the \( t + 1 \)th generation is a subset of the best individuals obtained by truncating the combined population of parents and offspring \( \mathcal{F}(t) + \mathcal{R}(t) \), i.e. \( |\mathcal{F}(t) + \mathcal{R}(t)| = 2\mu \) and \( |\mathcal{F}(t + 1)| = \mu \). To obtain \( \mathcal{F}(t + 1) \), \( \mathcal{F}(t) + \mathcal{R}(t) \) is first classified into non-dominated fronts. The first front \( \mathcal{F}_1 \) contains the Pareto optimal set \( \mathcal{P}_1 \) of \( \mathcal{F}(t) + \mathcal{R}(t) \). The subsequent fronts \( \mathcal{F}_j, j > 1 \), contain lower level non-dominated solutions and are obtained by disregarding solutions corresponding to the previous higher non-dominated fronts, i.e. \( \mathcal{F}, j > 1 \), is obtained from the set \( \mathcal{F}(t) + \mathcal{R}(t) - \bigcup_{k=1}^{j-1} \mathcal{F}_k \). Once the classification of non-dominated fronts is over, the parent population \( \mathcal{F}(t + 1) \) is filled with solutions belonging to the higher fronts, starting with front \( \mathcal{F}_1 \). If the whole front \( \mathcal{F} \), does not fit, the required number of individuals with best crowding distance are selected to fill the parent population. Each solution in \( \mathcal{F}(t + 1) \) is assigned a rank (fitness) equal to its non-domination level, where 1 is the best level. Binary tournament selection with crowded tournament operator, recombination, and mutation operators are used to create the offspring population \( \mathcal{R}(t + 1) \) from \( \mathcal{F}(t + 1) \). During selection, solution \( x \) wins a tournament if it has a better rank than \( y \). If \( x \) and \( y \) have the same rank, the solution with best crowding distance wins. The reader is referred to Deb, 2000 for additional details.

4.2. SPEA2

SPEA2 (Zitzler et al., 2001) creates a population \( \mathcal{E}(t) \) of individuals by genetic operators at the \( t \)th generation and introduces elitism by maintaining and external population \( \mathcal{E}(t) \) that stores a fixed number of the best solutions found since the beginning of the simulation. Note that the sizes \( |\mathcal{F}(t)| \) and \( |\mathcal{E}(t)| \) of the populations could be different. Each member \( i \) in the external population \( \mathcal{E}(t) \) and population \( \mathcal{F}(t) \) is assigned a strength \( s(i) \), representing the number of solutions it dominates in the combined population \( \mathcal{F}(t) + \mathcal{E}(t) \). An initial raw rank \( g_i \) for individual \( i \) is determined by the sum of strengths of its dominators, i.e. \( g_i = \sum s(j) \), where \( j \in \mathcal{F}(t) + \mathcal{E}(t) \) and \( j \) dominates \( i \). Note that fitness is minimized here. In addition, SPEA2 uses information about the density of solutions to discriminate between individuals having identical raw rank. Density \( d(i) \) of an individual \( i \) is calculated as the inverse of the distance to the \( k \)th nearest neighbor. Specifically, \( d(i) = 1/(\sigma_i^k + 2) \), where \( \sigma_i^k \) is the distance in objective space to the \( k \)th closest individual, and \( k = \sqrt{|\mathcal{F}(t)| + |\mathcal{E}(t)|} \). Adding \( d(i) \) to \( g(i) \) yields the final rank of the individual, i.e. \( f(i) = g(i) + d(i) \). The archive for the next generation \( \mathcal{E}(t + 1) \) consists of Pareto optimal solutions from \( \mathcal{F}(t) + \mathcal{E}(t) \), truncating \( \mathcal{E}(t + 1) \) in case the number of non-dominated solutions exceeds the specified size of the external population. If Pareto optimal solutions from \( \mathcal{F}(t) + \mathcal{E}(t) \) do not fill the external
population, then \( \delta(t+1) \) can also contain highly ranked dominated solutions from these populations. The population for the next generation \( Q(t+1) \) is created from \( \delta(t+1) \) by applying binary tournament selection with replacement, recombination, and mutation. The truncation method for the external population \( \delta(t) \) guarantees the preservation of boundary solutions. For additional details the reader is referred to Zitzler et al. (2001).

4.3. moRBC(\( c:1+1 \))

moRBC(\( c:1+1 \)) is a multiobjective random one-bit climber that uses a population for restarts (Aguirre and Tanaka, 2004b). moRBC(\( c:1+1 \)) at all times keeps one parent individual from which it creates one offspring. It begins with a randomly created parent string of length \( N \). Then, a random permutation \( \pi \) of the string positions is generated. A child is created by cloning the parent and flipping the bit at position \( \pi_i \). Next, the child is evaluated and replaces the parent if it dominates the parent. Child creation, evaluation, and (possibly) parent replacement are repeated for all positions in the random permutation, i.e. for all \( \pi_i, 1 \leq i \leq N \). After this, if parent replacements were detected testing continues by going back to create a new permutation \( \pi \). If no parent replacements were detected a dominance local optimum has been found and moRBC(\( c:1+1 \)) restarts the search. This process ends once a given number of evaluations has been expended. Fig. 10 illustrates the flow of moRBC(\( c:1+1 \)). A Population of up to \( \gamma \) solutions non-dominated by the parent and amongst themselves are kept during the process. moRBC(\( c:1+1 \)) restarts the search by replacing the parent with one individual chosen from the collected Population. If Population is empty, the parent is replaced with a random string created anew. Additionally, the non-dominated solutions found throughout the search are kept in an Archive of limited capacity. The procedures that update the Population and the Archive use NSGA-II’s diversity preserving mechanism in objective space, where non-dominated individuals with better crowding distance (Deb, 2001, p. 236) are preferred in case the Population/Archive has reached its capacity. Duplicate solutions are not allowed in the Population or in the Archive.

5. Metric, test problems, and parameters

In this work we use the hypervolume metric \( H \) proposed by Zitzler (1999) to evaluate and compare the performance of the algorithms. Let \( P_1 \) be a set

Step 1 Initialize the single parent \( p \) with a random string of length \( N \), add \( p \) to Archive, and set the number of evaluations counter to \( t = 1 \)

Step 2 While \(( t < T )\)

Step 2.1 Create a random permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_N) \) of the \( N \) string positions, set the permutation index to \( i = 1 \), and set local optimum to yes

Step 2.2 While \(( i < N \) and \( t < T )\)

- Clone the parent \( p \) and flip the bit at position \( \pi_i \) to create child \( c \). The child \( c \) is a one-bit neighbor of \( p \)
- If \( c \) dominates \( p \), replace \( p \) with \( c \) and set local optimum to no
- If \( c \) is not dominated by \( p \), update Archive and Population with \( c \)
- Increment \( t \) and \( i \), \( t = t + 1 \) and \( i = i + 1 \)

Step 2.3 If local optimum is yes, restart the search by replacing \( p \) with an individual of the Population. If Population is empty, \( p \) is initialized anew with a random string. Increment \( t, t = t + 1 \)

Step 3 Return Archive.

Fig. 10. The flow of multiobjective bit climber moRBC(\( c:1+1 \)).
of non-dominated solutions. The metric $H$ calculates the volume of the $M$-dimensional region in objective space enclosed by the elements of $P_1$ and a dominated reference point, hence computing the size of the region $P_1$ dominates, as illustrated in Fig. 11. The hypervolume can be expressed as

$$H(P_1) = \frac{\left| \bigcup_{i=1}^{\left| P_1 \right|} (F_i - \bigcap_{j=1}^{i-1} F_j) \right|}{C_0^i},$$

where $F_i$ is the hypervolume rendered by the fitness values of the $i$th solution $x_i \in P_1$ and the reference point. The hypervolume is among the few recommended metrics for comparing non-dominated sets (Knowles and Corne, 2002) and there is some theoretical evidence (Fleischer, 2003) that the maximization of the hypervolume constitutes the necessary and sufficient condition for the solutions in objective space to be maximally diverse Pareto optimal solutions of a discrete, multiobjective, optimization problem. The reference point to calculate the hypervolume is set to $[0.0, \ldots, 0.0]$.

In the following sections we use MNK-landscapes with $M = \{2, 3, 5\}$ objectives, $N = \{20, 50, 100\}$ bits, vary the number of epistatic interactions from 0% to 50% of $N$ simultaneously in all objectives ($K_1, \ldots, K_M = K$), and set random epistatic patterns among bits for all objectives ($D_1, \ldots, D_M = \text{random}$). For each combination of $M$, $N$ and $K$, 50 different problems randomly generated are employed.

NSGA-II and SPEA2 use a population size of 100 individuals, two-point crossover for recombination with rate $p_c = 0.6$, and bit flipping mutation with rate $p_m = 1/N$ per bit. moRBC($\gamma = 1 + 1$) also uses a population for restarts $\gamma = 100$ individuals. For all algorithms, the number of evaluations is set to $3 \times 10^5$ and the Archive size is set to 100.

6. Performance by conventional NSGA-II and SPEA2

First, we present results by conventional NSGA-II and SPEA2 on scalable MNK-landscapes for various values of $M$, $N$, and $K$, in order to have a broad view of the performance of these algorithms on combinatorial multiobjective epistatic problems. Fig. 12 plots the Archive’s average hypervolume over the number of epistatic interactions $K$ for $N = \{20, 50, 100\}$ bits landscapes. The average hypervolume of the true Pareto front obtained by enumeration is also included for $N = 20$ bits landscapes. Vertical bars overlaying the mean hypervolume curves represent 95% confidence intervals.

From Fig. 12(a), note that on $N = 20$ bits landscapes the trend of the hypervolume of the true Pareto front for any value of $M$ is to rapidly increase with $K$, from $K = 0$ to small values of $K$, and to remain high for medium and large $K$. A similar trend is expected for the hypervolume of the true Pareto front on landscapes with higher values of $N$. Looking at results by NSGA-II and SPEA2, in Fig. 12(a) we can see that the hypervolume of the solutions found by these algorithms approach the hypervolume of true Pareto front on $N = 20$ bits landscapes only for $K \leq 15\% N$. Increasing the number of bits $N$, we see that the value of the hypervolume of the non-dominated solutions found by the algorithms decreases continuously from $K \geq 8\% N$ for $N = 50$ and from $K \geq 5\% N$ for $N = 100$ bits. See Fig. 12(b) and (c), respectively. These decreasing values are against the expected trend of the hypervolume of the true Pareto front and indicate that the search performance of the algorithms is worsening significantly as $K$ increases.

Although there are some differences on the implementations of NSGA-II and SPEA2, both algorithms are designed based on the same principles and their performance trends are similar. In the following, for the sake of clarity, we focus on NSGA-II and especially look in detail into the effects of selection, drift, recombination, and mutation.
7. Selection and drift

The main processes that drive evolution are selection, drift, mutation, and recombination. In this section we observe the effects of selection and drift, which decrease genetic variation. An extreme consequence of decreasing genetic variation is the homogenization of the population. Selection features are made explicit during the design of the algorithms. In most of the latest generation MOEAs selection incorporates elitism and it is biased by Pareto dominance and a diversity preserving strategy in objective space. On the other hand, drift is a process implicit to all stochastic algorithms working on finite small populations.

7.1. Elitism

Elitism is considered an important component of the selection process in state of the art evolutionary multiobjective optimizers (Deb, 2001; Coello et al., 2002). In order to have a clear idea of its contribution to the performance of MOEAs on epistatic problems this section compare results by NSGA-II(\(\mu + \lambda\)) and moGA(\(\mu, \lambda\)). NSGA-II implements elitism by keeping for the next generation the best \(\mu\) individuals from the joined population \(\lambda\) of parents and offspring. On the other hand, moGA(\(\mu, \lambda\)) replaces the parent population \(\lambda\) by its offspring population \(\lambda\) at each generation. moGA(\(\mu, \lambda\))’s other features are the same used by NSGA-II.

Fig. 13 shows results by NSGA-II(100 + 100) and moGA(100,100) for \(M = \{2, 3, 5\}\) objectives on \(N = 100\) bits landscapes. From this figure we can see that if elitism is not included there is a severe deterioration in performance for all values of \(K\) and \(M\), except for \(M = 5\) and \(K = 5\). Note that the performance by moGA(100,100) falls sharply from \(K = 10\%\ N\) to \(K = 15\%\ N\) in \(M = 2\) objectives. The fall in performance is even more pronounced for \(M = 3\) and \(M = 5\) objectives. These results are in accordance with the expectation that elitism is a very important feature for multiobjective combinatorial optimization. However, elitism can also bring about undesired side effects that could severely affect the efficacy and efficiency of the algorithms. Throughout the following sections we discuss some of them.

7.2. Genetic drift

Genetic drift is a phenomenon that emerges from the stochastic operators of selection, recombination, and mutation. It refers to the change on bit (allele)
frequencies due to chance alone especially in small populations. In single objective EAs it is well known that genetic drift is one important factor that affects negatively the performance of EAs especially if a strong selection pressure is used, such as truncated selection \((\mu + \lambda)\) without elitism where \(\mu < \lambda\). See Aguirre and Tanaka (2003), for example. The presence of elitism, for instance in the form of truncated selection \((\mu + \lambda)\) used by NSGA-II, would increase selection pressure making elitist algorithms even more prone to the effects of drift. Another important cause of drift is the different magnitude of contribution to fitness by individual bits (Thierens et al., 1998).

In this section we enhance NSGA-II by preventing fitness duplicates from the population in order to observe the effect of genetic drift on the performance of the algorithm. In the enhanced algorithm, called NSGA-II ed, if several individuals have exactly the same fitness in all objectives then one is chosen at random and kept. The other equal fitness individuals are eliminated from the population. Fitness duplicates elimination is carried out before truncating the population from \((\mu + \lambda)\) to \((\mu)\) individuals. Another important cause of drift is the different magnitude of contribution to fitness by individual bits (Thierens et al., 1998).

In this section we enhance NSGA-II by preventing fitness duplicates from the population in order to observe the effect of genetic drift on the performance of the algorithm. In the enhanced algorithm, called NSGA-II ed, if several individuals have exactly the same fitness in all objectives then one is chosen at random and kept. The other equal fitness individuals are eliminated from the population. Fitness duplicates elimination is carried out before truncating the population from \((\mu + \lambda)\) to \((\mu)\) individuals. This process aims to effectively eliminate clones without the need to compare Hamming distances, postpone genetic drift, and remove an unwanted source of selective bias.

To explain and understand better the effects of duplicates on the performance of multiobjective algorithms, Fig. 14 shows cumulative bar diagrams of the average number of individuals per non-dominated front over the number of epistatic interactions \(K\). Results by the conventional NSGA-II and by the enhanced NSGA-II ed that eliminates duplicates are presented for the top five fronts in \(M = \{2, 3, 5\}\) objectives. In the case of conventional NSGA-II, the figure also shows with lines the number of duplicate individuals \(d\) in the whole population \((\mu + \lambda)\) before truncation, in the first non-dominated front \((\mathcal{F}_1)\) also before truncation, and in the truncated population \((\mu)\). For NSGA-II ed results are presented after elimination of duplicates. Horizontal lines indicate the truncation site.

Looking at results by NSGA-II in Fig. 14, the following observations are relevant. (i) The number of duplicates increases as we increase the epistatic interactions \(K\). (ii) The presence of duplicates reduces when the number of objectives \(M\) increases. (iii) Most duplicates belong to the first non-dominated front and a large number remain after truncation, especially for large \(K\). Conversely, looking at the size of the cumulative bars by NSGA-II ed in Fig. 14, we can deduce that the average number of duplicates eliminated at each generation by NSGA-II ed is only a small fraction of the whole \((\mu + \lambda)\) population and it is similar for all \(K\) and \(M\). For example, for \(M = 2\) note that in NSGA-II
the number of duplicates augment from 8% to 90% of the truncated population ($\mu$) and from 18% to 60% of the whole population ($\mu + \lambda$) increasing $K$ from 0 to 50, respectively. By contrast, the average number of duplicates in NSGA-IIed is around 9% for all $K$. We should clarify that genetic diversity is important even if the individuals have the same fitness. However, in the results we report here all fitness duplicates for any value of $K$ and $M$ are actually clones. It should be also said that in single objective NK-landscapes 99.88% of fitness duplicates are clones (Aguirre and Tanaka, 2003). The number of duplicates observed in NSGA-IIed could be taken as the homogenization effect of drift and selection at each generation, whereas the number of duplicates in NSGA-II should be taken as the amplified effect of drift and selection throughout the generations.

Duplicates hinder exploration and selection as well. If duplicates are not eliminated at each generation they accumulate rapidly decreasing the likelihood that the algorithm will explore a larger number of different candidate solutions during a run. Also, since the chances of selecting a given genotype are multiplied by the number of clones of that genotype present in the population, duplicate genotypes end up with higher selective advantage than unique genotypes. This unwanted selective bias is not based in actual fitness and cannot be avoided by ranking procedures, scaling mechanism, or even truncated deterministic mechanisms. A reduced explorative capability combined with an unwanted selective bias can considerably affect the possibility of finding better non-dominated solutions.

Fig. 15 shows the hypervolume by NSGA-IIed(100 + 100) that eliminates duplicates and by the conventional NSGA-II(100 + 100) to illustrate the effect of duplicates on the performance of the algorithms. From this figure, we can see that elimination of duplicates improves the performance of NSGA-II in two and five objective landscapes for $K \geq 5$ and $3 \leq K \leq 35$, respectively. In three objective landscapes we see almost no improvement by eliminating fitness duplicates. Note that the largest overall performance difference between NSGA-IIed and NSGA-II is for $M = 2$ objective landscapes, where precisely the accumulation of duplicates in the conventional NSGA-II is the highest as shown in Fig. 14. Now, since more duplicates are observed for $M = 3$ than $M = 5$ objectives, an interesting question is why do we see almost no improvement for $M = 3$ although we see it for $M = 5$? We answer this question in the next section.

![Fig. 14. Average number of individuals per front and average number of duplicates by conventional NSGA-II and enhanced NSGA-IIed. $N = 100$ bits. (a) $M = 2$, and (b) $M = 3$ and (c) $M = 5$.](image)
The number of epistatic interactions $K$ and the number of objectives $M$ are important factors that determine the density of non-dominated fronts in multiobjective landscapes, as shown in Section 3. First, the number of non-dominated solutions in the top non-dominated fronts reduces as $K$ increases. Second, fixing the bit string length (size of the search space) and increasing the number of objectives $M$ we have landscapes with fewer but more dense non-dominated fronts. Third, the effect on front’s density by $M$ is stronger than the effect by $K$. These properties are clearly reflected in the distribution of solutions per front by NSGA-II ed in Fig. 14. Note that for NSGA-II ed the number of individuals in the first front decreases as $K$ increases. However, increasing $M$ the number of individuals in the first front increases and fewer fronts fit in the population.

A consequence of front’s density is that it could restrain selection, especially during mating. Taking as an example the way selection is done in NSGA-II, see Section 4.1, Pareto non-domination level will be important for mating mostly in $M = 2$ objectives where several fronts fit within the truncated population, except for $K = 0$. Increasing $M$, fewer but more dense fronts would increase the relative importance of crowding of solutions within a front over non-domination level as criterion to bias selection, especially if most individuals within the truncated population belong to the same front.

From the same Fig. 14, we can see that in fact the truncated population of NSGA-II ed would mostly come from the first front for $M = \{3, 5\}$. Thus, in both cases during mating the criterion to bias selection would be mainly crowding factor since most solutions would be ranked with the same non-domination level. However, there is a difference between $M = 3$ and $M = 5$ given by truncation. Truncation would reinforce diversity in objective space by purging individuals with high crowded factor if the number of individuals in the front is larger than the parent population size. Note that for $M = 3$ in NSGA-II ed the number of individuals in the first front $F_1$ is close to the size of the truncated population ($\mu$), for most $K$, and thus the algorithm does not have a chance to purge highly crowded individuals. On the other hand, for $M = 5$ the number of individuals in the first front is greater than the truncated population size ($\mu$), for all $K$, and truncation can contribute purging highly crowded individuals. This suggests that the increase in performance for $M = 5$ but not for $M = 3$ seems to be due to truncation acting more often on the first front for $M = 5$.

7.3. Selection bias by front level and objective space diversity

The number of epistatic interactions $K$ and the number of objectives $M$ are important factors that

---

![Fig. 15. Effect of drift on performance. $\mathcal{F}$ over $K$ by conventional NSGA-II and NSGA-II ed that eliminates fitness duplicates. $N = 100$ bits. (a) $M = 2$, (b) $M = 3$ and (c) $M = 5$.](image-url)
8. Recombination and mutation

To function effectively EAs must balance the processes of evolution that decrease genetic variation with those that increase it (Holland, 1975). In previous sections we have restricted our discussion to selection and drift, which decrease genetic variation. In the following we focus on the effectiveness of recombination and mutation, mechanisms that increase genetic variation. We also discuss issues that hinder exploration under elitist selection and try ways to make mutation more effective.

8.1. Crossover operator

In this section we observe the effect on performance of (not) using crossover. Fig. 16 shows results by NSGA-II ed(100 + 100) and M ed(100 + 100). M ed is an NSGA-II ed algorithm with crossover turned off using mutation as the sole variation operator, i.e. \( p_c = 0.0 \) and \( p_m = 1/N \) per bit. From Fig. 16 note that NSGA-II ed that includes crossover and mutation performs better than M ed that uses only mutation for \( K \leq 1 \), \( K \leq 3 \), and \( K \leq 7 \) for \( M = \{2, 3, 5\} \) objectives, respectively. For other values of \( K \) we do not see any contribution to performance by including crossover. In fact, we can see that mutation alone performs better for some values of \( K \), especially in landscapes with \( M = 2 \).

Results by crossover are in accordance with the effects of epistasis on multiobjective landscapes. In Section 3, it is shown that for small values of \( K \) non-dominated solutions of top fronts are highly correlated in decision (genotype) space, in objective (phenotype) space, and between spaces. However, this correlation decreases rapidly by increasing \( K \) and it decreases faster for smaller \( M \). For small \( K \), recombination of high fitness individuals would likely produce high fitness offspring. However, as the number of epistatic interactions \( K \) increases the likelihood that offspring would be far from the parents in objective space also increases considerably. The properties of MNK-landscapes offer no much hope for blind mating and recombination, i.e. just taking any two individuals from the best non-dominated front in the population and recombining them. In the literature there are some reports suggesting that mating based on proximity in decision or objective space could help recombination in MOEAs (Hajela and Lin, 1992; Fonseca and Fleming, 1993; Watanabe et al., 2002; Kim et al., 2004; Ishibuchi and Shibata, 2003a,b, 2004). It will be interesting to assess in the future the benefit of these approaches on scalable epistatic landscapes. How helpful are they as we increase \( K \)?
8.2. Elite’s age and mutation explorative range

An undesired side effect of elitism combined with a short explorative range by crossover and/or mutation is cyclically exploring same points. To explain this it is useful to see the crossover rate $p_c$ governing the application of two operators. One is crossover followed by mutation ($p_c$) and the other one is mutation alone ($1 - p_c$). In this section we focus on elitism and mutation and do not consider the case of crossover followed by mutation.

Conventional NSGA-II, for example, uses a $(\mu + \lambda)$ selection where elite solutions could remain in the population indefinitely. Additionally, mutation rate is often set to $p_m = 1/N$, which means that mutation will explore solutions in average one bit away from the parent in decision space. In this case, eventually after some generations offspring created from elite solutions would likely not be different from offspring created before, even in the case of perfect sampling (no drift). The expected time for mutation to start sampling again same points from an elite individual would be a function of the number of occasions the same elite individual has been selected for reproduction, the rate at which mutation is applied alone given by the probability $1 - p_c$, and the bit string length $N$.

A way to avoid this undesired cycles and enhance exploration is to put an age limit to elite solutions and bias selection accordingly. To observe the effects of age of elite solutions we create $M^{edo}$ from NSGA-IIed, $M^{edo}$ increases by one the age of an elite solution each time it is selected for reproduction, the rate at which mutation is applied alone given by the probability $1 - p_c$, and the bit string length $N$.

A way to avoid this undesired cycles and enhance exploration is to put an age limit to elite solutions and bias selection accordingly. To observe the effects of age of elite solutions we create $M^{edo}$ from NSGA-IIed, $M^{edo}$ increases by one the age of an elite solution each time it is selected for reproduction, the rate at which mutation is applied alone given by the probability $1 - p_c$, and the bit string length $N$.

We also verify whether a local search-like strategy would be more effective than the conventional bit flipping mutation strategy. To do that we create $aM^{edo}$ from $M^{edo}$, $aM^{edo}$, in addition to eliminating duplicates and very old elite individuals, it also uses the age to guide mutation. The bit string of length $N$ is subdivided in $S$ segments of length $L$, $N = S \times L$. For elite individuals, age greater than one, mutation flips one bit at the position indicated by $j + i$, $j = L \times \lfloor \text{rand()} \mod S \rfloor$ and $i = \lfloor \text{age} \mod N \rfloor \mod L$, i.e. the mutation segment is chosen at

![Fig. 17. Elite’s age and mutation explorative range.](image-url)
random and the bit within the segments is given by the age of the individual. The main objective of this kind of mutation is to increase the chances of exploring most one-bit neighbors of an elite individual as its age approaches \( N \). Also, it makes sure that only one bit will be flipped. Note that using mutation guided by age we do not need to keep a memory for each mutated bit in each individual. In our experiments \( N = 100 \) and \( S = L = 10 \). Standard flipping mutation is applied for individuals whose age is one \( (p_c = 0.0) \).

Fig. 17 shows results by the mutation-only algorithms \( M^{redo}(100 + 100) \) and \( aM^{redo}(100 + 100) \) together with results by NSGA-II ed. Looking at results by \( M^{redo} \) we can see that preventing old elite individuals increases substantially the performance of NSGA-II ed for all number of objectives \( M \) and most values of epistatic interactions \( K \). Note that in this case there is no more a performance advantage offered by crossover in small \( K \) landscapes, except for \( M = 5 \) and \( K \leq 1 \). Looking at results by \( aM^{redo}(100 + 100) \) also note that eliminating old elite individuals combined with local search-like mutation strategy informed by age further improves performance, especially for medium and high \( K \) as we increase \( M \).

9. Comparison with moRBC

In this section we compare the performance of conventional NSGA-II, the enhanced algorithm \( aM^{redo} \), and the population-based multiobjective
random one-bit climber moRBC(γ; 1 + 1) (Aguirre and Tanaka, 2004b). Results are shown in Fig. 18. From this figure we can see that the performance of conventional NSGA-II is worse by several standard deviations than the performance of moRBC(γ; 1 + 1), for all values of K and M. In contrast, note that the performance of the enhanced algorithm aM eed approaches the performance of moRBC(γ; 1 + 1) in M = 2 and M = 3 objectives. However, for M = 5 objectives moRBC(γ; 1 + 1) still performs better. These results reveal that conventional elitist Pareto-based MOEAs with crossover and standard bit flipping mutation perform overall very poorly on MNK-landscapes of increased levels of epistatic interactions. Based on the knowledge obtained in this work, we should look for ways to design and implement strong and robust MOEAs that can achieve good performance on landscapes of increased epistatic complexity. Also, in addition to MNK-landscapes, we should investigate the performance of MOEAs on problems with other forms of epistatic interactions.

10. Conclusions

In this work we have studied the working principles, behavior, and performance of MOEAs on multiobjective epistatic fitness functions with discrete binary search spaces by using MNK-landscapes. First, we have analyzed the structure and some of the properties of MNK-landscapes under a multiobjective perspective by using enumeration on small landscapes. Then, we have studied the effects of selection, drift, recombination (crossover), and mutation in MOEAs on large MNK-landscapes. We have shown that enhancing selection and postponing drift by eliminating fitness duplicates and removing old elite individuals helps to increase substantially the performance of MOEAs. We also observed that recombination adds to the performance of standard bit flipping mutation only for small values of epistatic interactions. However, any gain by crossover is largely surpassed by doing a more effective exploration with short-ranged mutation alone. Enhancements in selection, postponing drift, and explorative efficiency have considerably increased the robustness of MOEAs across several sub-classes of epistatic MNK-landscapes problems and number of objectives. Yet, these enhancements are not enough to surpass the performance of simpler population-based multiobjective random one-bit climbers (Aguirre and Tanaka, 2004b) and we should look for ways to design better MOEAs. Results in this work strongly suggest that elitism combined with an efficient short-range explorative capability by mutation is highly effective and likely to be a required feature of MOEAs. The advantages, if any, of elitism combined with mutation using larger explorative ranges should be investigated in the future. Also, we should look into the influence of population size, other variation operators, and special mating strategies for recombination to further clarify its role in multiobjective discrete binary search spaces. Moreover, we should investigate the performance of MOEAs on problems with other forms of epistatic interactions in addition to MNK-landscapes.

References


