A Niching Scheme for EDAs to Reduce Spurious Dependencies

Po-Chun Hsu
Tian-Li Yu

TEIL Technical Report No. 2013001
March, 2013

Taiwan Evolutionary Intelligence Laboratory (TEIL)
Department of Electrical Engineering
National Taiwan University
No.1, Sec. 4, Roosevelt Rd., Taipei, Taiwan

http://teil.ee.ntu.edu.tw
A Niching Scheme for EDAs to Reduce Spurious Dependencies

Po-Chun Hsu, Tian-Li Yu
Department of Electrical Engineering
National Taiwan University
r00921034@ntu.edu.tw, tianliyu@cc.ee.ntu.edu.tw

Abstract

This paper proposes a niching scheme, the dependency structure matrix restricted tournament replacement (DSMRTR). The restricted tournament replacement (RTR) is a well-known niching scheme in the field of estimation of distribution algorithms (EDAs). However, RTR induces spurious dependencies among variables, which impair the performance of EDAs. This paper utilizes building-block-wise distances to define a new distance metric, the one-niche distance. For those EDAs which provide explicit linkage information, the one-niche distances can be directly incorporated into RTR. For EDAs without such information, DSMRTR constructs a dependency structure matrix via the differential mutual complement to estimate the one-niche distances. Empirical results show that DSMRTR induces fewer spurious dependencies than RTR does while maintaining enough diversity for EDAs.

1 Introduction

Estimation of distribution algorithms (EDAs) (Larranaga & Lozano, 2002; Goldberg, 2002) construct dependency models from promising individuals and utilize the information gained from the models to generate a new population of individuals. Two types of inaccuracies in EDA models have been studied: missing dependencies and spurious dependencies. While most earlier work focused on the identification of important dependencies (Etxeberria & Larrañaga, 1999; Harik, Lobo, & Sastry, 2006; Pelikan, Goldberg, & Cantú-Paz, 1999; Yu, Goldberg, Yassine, & Chen, 2003; Tsuji, Munetomo, & Akama, 2004; Yu & Goldberg, 2004), recent studies (Yu, Sastry, & Goldberg, 2005; Santana, Larrañaga, & Lozano, 2005; Lima, Pelikan, Sastry, Butz, Goldberg, & Lobo, 2006; Lima, Pelikan, Goldberg, Lobo, Sastry, & Hauschild, 2007; Radetic & Pelikan, 2010; Lima, Lobo, Pelikan, & Goldberg, 2011) have argued that spurious dependencies impair the performance of EDAs. Some of these studies are described as follows.

Yu et al. (Yu, Sastry, & Goldberg, 2005) used a qualitative model to show that spurious dependencies inhibit EDAs from effectively mixing partial solutions. Lima et al. (Lima, Pelikan, Sastry, Butz, Goldberg, & Lobo, 2006) found that spurious dependencies in models of BOA (Pelikan, Goldberg, & Cantú-Paz, 1999) impair the performance of efficiency-enhancement techniques that depend on model accuracy such as substructural hillclimbing. Radetic and Pelikan (Radetic & Pelikan, 2010) empirically showed that spurious dependencies impair the performance and scalability of ECGA (Harik, Lobo, & Sastry, 2006) on the OneMax problem.

Another important issue in the EDA field is whether or not enough diversity of the population is maintained. A niching scheme preserves diverse solution candidates to prevent premature convergence. The restricted tournament replacement (RTR) (Harik, 1995; Pelikan & Goldberg, 2001) is a commonly used niching schemes in EDAs. EDAs such as hBOA (Pelikan & Goldberg, 2001) and DSMGA (Yu, Goldberg, Yassine, & Chen, 2003) have used RTR to solve problems with hierarchy and overlap. Although RTR has been shown to maintain multiple niches, researches (Lima, Pelikan, Goldberg, Lobo, Sastry, & Hauschild, 2007; Radetic & Pelikan, 2010) pointed out that it induces more spurious dependencies than other replacement methods do.

The purpose of this paper is to develop a new replacement scheme that induces fewer spurious dependencies than RTR does and maintains enough diversity of the population. RTR distorts the
Algorithm 1: RTR

Input:
- The original population, \( X = \{ \vec{x}_0, \vec{x}_1, \ldots, \vec{x}_{n-1} \} \);
- The offspring population, \( Y = \{ \vec{y}_0, \vec{y}_1, \ldots, \vec{y}_{n-1} \} \);
- The window size, \( w \) (\( w \leq n \));

Output:
- The population \( X \) for the next generation;

1. for each \( \vec{y} \in Y \) do
2. Choose a random subset \( S \subset N \) consisting of \( w \) integers, where \( N = \{ 0, 1, \ldots, n-1 \} \);
3. Let \( m \) be any element in \( \underset{\vec{x} \in S}{\text{argmin}} \text{Distance}(\vec{x}, \vec{y}) \);
4. if \( \text{Fitness}(\vec{y}) > \text{Fitness}(\vec{x}_m) \) then
5. \( \vec{x}_m \leftarrow \vec{y} \);
6. return \( X \);

distribution of individuals more than necessary. This paper confirms that the distortion is a cause of spurious dependencies and focuses on resolving this problem.

The rest of this paper is structured as follows. The next section presents how it excessively distorts the distribution of individuals. Section 3 defines a new distance metric to refine RTR with regard to the model accuracy. The modified version of RTR can be directly incorporated into EDAs which utilize explicit linkage groups to represent building blocks (BBs (Goldberg, 2002)). For EDAs which do not generate explicit linkage groups, Section 4 proposes the dependency structure matrix restricted tournament replacement (DSMRTR). DSMRTR utilizes a dependency structure matrix (DSM) to estimate the distances without the explicit linkage information. In Section 5, experiments are conducted to test this new niching scheme. Finally, this paper concludes with a summary and possible future work.

2 RTR and Its Drawback

This section describes the mechanism of RTR and presents how it distorts the distribution of individuals. The excessive distortion is the main reason for our modification.

2.1 How RTR Preserves Niches

Algorithm 1 presents the process of RTR, where the function \( \text{Distance} \) in Line 3 is usually the Hamming distance function. Harik suggested (Harik, 1995) that RTR preserves a class of sets of solution candidates called \textit{optimal sets} and described how it works as follows. Here presents the definition of optimal sets.

\textbf{Definition 1} Let \( Z \) be the search space, \( d : Z \times Z \to \mathbb{R} \) be a distance measure over \( Z \), \( f : Z \to \mathbb{R} \) be the fitness function and \( S \subset Z \) be a set of solution candidates. \( S \) is an \textbf{optimal set} under \( d \) if

\[ \forall \vec{y} \in Z : \left( \forall \vec{m} \in \arg\min_{\vec{x} \in S} d(\vec{x}, \vec{y}) : f(\vec{y}) \leq f(\vec{m}) \right). \] (1)

Consider the situation where an EDA has placed an equal number of points at \( r \) optimal peaks throughout the search space, and these locally optimal solutions compose an optimal set. When a new individual \( \vec{y} \) is generated, it has to compete with the closest one out of a subset chosen from the population. The size of the subset is called the window size. Let the window size be \( w \), where \( w = cr \). Assume that \( \vec{s}_0 \) is the closest element to \( \vec{y} \) in the optimal set and that copies of \( \vec{s}_0 \) occupy a proportion of \( \frac{r}{w} \) in the population. The fitness value of \( \vec{y} \) is less than or equal to that of \( \vec{s}_0 \). The probability that no copies of \( \vec{s}_0 \) are present among the \( w \) chosen elements can be approximated as

\[ (1 - \frac{1}{r})^w = (1 - \frac{1}{r})^{cr} < e^{-c}. \] (2)
If the window size is much greater than the number of the optimal solutions to be preserved, there is a small probability that any element out of the optimal solutions is replaced by a newly generated individual.

The above is the argument that RTR maintains a set of locally optimal solutions during the replacement procedure. Pelikan (Pelikan, 2005) suggested a window size equal to the problem size. We use the window size that equals the problem size but is bounded to the population size in all the experiments in this paper.

2.2 RTR and Induced Spurious Dependencies

Although RTR successfully preserves diverse individuals by forcing a newly generated individual to compete against a similar individual, this mechanism inevitably changes the underlying distribution of the individuals in the population (Harik, 1995). If the distortion of the distribution is excessive, it implicitly increases the difficulty of a problem and hampers the performance of EDAs.

Let $\eta$ satisfy $0 \leq \eta \leq 1$ to be a proportion and $n$ be the population size. In RTR, for an offspring individual, the probability that an individual chosen to compete lies among the closest $\eta \cdot n$ individuals in the population can be expressed as (Harik, 1995)

$$g(\eta) = 1 - (1 - \eta)^w.$$ (3)

Taking the derivative of this function with respect to $\eta$, one gets

$$\frac{\partial g(\eta)}{\partial \eta} = w(1 - \eta)^{w-1}. \quad (4)$$

The derivative decreases as $\eta$ increases in the whole domain $0 \leq \eta \leq 1$. This indicates that between two competitions, RTR prefers the competition where the two individuals are closer even when neither of the competitions occurs within a niche. This type of discrimination is redundant with regard to preserving niches. In short, RTR distorts the distribution of the individuals more than necessary.

The biased distribution of individuals may cause EDA models to misidentify the dependencies among variables. Empirically, RTR impairs the model accuracy more than other replace methods do (Lima, Pelikan, Goldberg, Lobo, Sastry, & Hauschild, 2007; Radetic & Pelikan, 2010). In order to reduce distorting the distribution of individuals during the RTR procedure, the following section defines a new distance metric different from the commonly used Hamming distances.

3 New Distance Metric to Reduce Distortion of the Distribution

In this section, we define a distance metric to prevent RTR from excessively distorting the distribution of individuals. This metric utilizes BB-wise distances, which can be calculated from the explicit models provided by some EDAs. Experiments on the OneMax problem and the B-trap problem (Fan, Yu, & Lee, 2011) are conducted to test this approach.

3.1 One-Niche Distance Metric

For two solutions $\vec{x}$ and $\vec{y}$ in the search space, we define the one-niche distance as follows.

Definition 2 The one-niche distance between two solutions $\vec{x}$ and $\vec{y}$ is defined as

$$d_{OneNiche}(\vec{x}, \vec{y}) = \begin{cases} \infty & \text{if } d_{BB}(\vec{x}, \vec{y}) > 1 \\ d_H(\vec{x}, \vec{y}) & \text{otherwise}, \end{cases}$$ (5)

where $d_H$ is the Hamming distance function and $d_{BB}$ is the BB-wise distance function.
Competitions between individuals can be classified into two types. In an intra-niche competition, both individuals occupy the same niche; in an inter-niche competition, the two individuals occupy different niches. A classifier between these two types is required to reduce the discrimination between competitions of the same type.

Since Hamming distances reveal little about whether or not two individuals occupy the same niche, the proposed distance metric utilizes BB-wise distances. The BB-wise distance between two individuals is defined as the number of the BBs in which they differ. For instance, let \( \vec{x} = 1011 1111 1111 \) and \( \vec{y} = 0101 1101 1111 \) be two solutions to a 12-bit problem where each four adjacent bits compose a BB. One gets \( d_H(\vec{x}, \vec{y}) = 4 \) and \( d_{BB}(\vec{x}, \vec{y}) = 2 \).

Consider a decomposable problem consisting of \( m \) BBs and \( \chi \) optimal peaks within each BB. The number of optimal solutions to the problem is \( \chi^m \). If an EDA successfully learns the problem structure, it solves subproblems separately and maintains the niches within each BB separately (Pelikan & Goldberg, 2001; Sastry, Abbass, & Goldberg, 2004). An EDA thus needs to preserve only \( \chi \cdot m \) niches if the models are accurate. These \( \chi \cdot m \) niches are considered necessary while the other niches are considered unnecessary.

Assume that no more niches than necessary are preserved in the population. Two individuals that differ in more than one BB can be considered to occupy different niches. Therefore, if \( d_{BB}(\vec{x}, \vec{y}) \) is greater than one, the one-niche distance \( d_{OneNiche}(\vec{x}, \vec{y}) \) is set to be infinity no matter the number of BBs or bits that \( \vec{x} \) and \( \vec{y} \) differ in. This threshold for the BB-wise distance is the first classifier between intra-niche competitions and inter-niche competitions. Almost all pairs of individuals that occupy different niches are equivalent in the process of measuring similarities. This setting is the key to maintaining the distribution of individuals.

An EDA needs \( \chi \) niches for each subproblem, and hence individuals which differ in only one BB are not certainly in the same niche. The mechanism of the traditional RTR is still adopted in order to further prompt the competitions to occur within niches, if the BB-wise distance is one. This part slightly distorts the distribution of individuals, but the distortion is relatively negligible if the number of BBs is large enough.

We substitute the function Distance in Line 3 of Algorithm 1 with the one-niche distance function to form a modified version of RTR. The modified RTR preserves optimal sets, according to Definition 1 under the one-niche distance measure \( d_{OneNiche} \). The optimal sets cannot cover all the \( \chi^m \) optimal solutions. Nevertheless, they contain solution candidates out of the \( \chi \cdot m \) necessary niches, which are enough to solve the problem.

The modified RTR does not deviate from the original arguments described in Section 2.1. The one-niche distance between two individuals in the same niche is set to be less than that between two individuals which occupy different niches. Hence, the probability that any element out of the maintained niches is replaced by a newly generated individual is still upper bounded by \( e^{-c} \), as in Equation 2.

Some EDAs provide explicit linkage groups, such as ECGA and DSMGA. One can directly compute BB-wise distances from the linkage groups. Take ECGA for example. ECGA uses marginal product models (MPMs) to sample new solution candidates. An MPM in ECGA represents a disjoint partition of variables in the problem. Each set in the partition is a linkage group.

### 3.2 Spurious Dependencies on OneMax

The OneMax problem consists of maximizing the number of ones in a binary string. The fitness function of this problem is defined as

\[
fonemax(\vec{x}) = \sum_{i=1}^{\ell} x_i,
\]

where \( x_i \) is the \( i \)-th bit in the input binary string \( \vec{x} \) and \( \ell \) is the length of \( \vec{x} \). In this problem, bits are independent of each other. EDA models should not consist of linkage groups with sizes more than one, in order to maximize mixing (Thierens & Goldberg, 1993; Yu, Sastry, & Goldberg, 2005) and minimize the population-sizing requirements (Harik, Cantú-Paz, Goldberg, & Miller, 1997; Goldberg, Sastry, & Latoza, 2001; Yu, Sastry, Goldberg, & Pelikan, 2007). However, for ECGA on the OneMax problem, RTR was shown (Radetic & Pelikan, 2010) to induce more and
larger spurious linkage groups than FR and ER do. These spurious dependencies hamper the performance of ECGA.

We conduct an experiment to compare the modified RTR with FR, ER and the traditional RTR which utilizes Hamming distances. In FR, the offspring population completely replaces the original population. In ER, the worst 50% of the original population is replaced with the offspring population. The binary tournament selection is used as the selection operator. These settings are common in all the experiments described in this paper.

Figure 1 indicates that the modified RTR resolves the problem of spurious linkage groups for ECGA on the OneMax problem. The modified RTR does not induce larger linkage groups than FR and ER do. Since the design of the one-niche distance metric is based on preserving the distribution of individuals, this experiment also confirms that a cause of spurious dependencies is excessively distorting the distribution of individuals.

3.3 Spurious Dependencies on B-Traps

We conduct another experiment on a test problem where the strength of dependencies among variables is adjustable. This test problem consists of concatenated B-traps (Fan, Yu, & Lee, 2011). Let \( u(\vec{x}) \) be the number of ones in the binary string \( \vec{x} \). Given a real number \( \alpha \) in the interval \( 0 \leq \alpha \leq 1 \), a B-trap function of order \( k \) is defined as

\[
b_k(\vec{x}) = 0.9(1 - t)^2 + t^2,
\]

where \( t \) is one of the solutions to the quadratic equation

\[
(1 - 2\alpha)t^2 + 2\alpha t = \frac{u(\vec{x})}{k}
\]

that satisfies \( 0 \leq t \leq 1 \). In the B-trap function, \( \alpha \) is a parameter to adjust the degree of deception, as shown in Figure 2. When \( \alpha \) gets closer to one, the B-trap function becomes more similar to a trap function of order \( k \) (Deb & Goldberg, 1993); otherwise, the B-trap function becomes less deceptive. The fitness function of the problem composed of \( m \) concatenated \( k \)-bit B-traps is defined
Figure 3: NFE required for ECGA to solve the 100-bit problem composed of 20 concatenated B-traps of order 5 ($\ell = 100$) with different values of $\alpha$.

as

$$f_{B\text{trap}}(\vec{x}) = m - \sum_{i=0}^{m-1} b_k(x_{ik+1}x_{ik+2} \cdots x_{ik+k}).$$

(9)

We use B-traps with different values of $\alpha$ to verify whether the modified RTR reduces the spurious dependencies and maintains enough diversity of the population. Figure 3 presents the required number of function evaluations (NFE) for ECGA to solve these problems.

The B-traps are highly deceptive when $\alpha$ is close to one. The modified RTR consumes fewer NFE than FR and ER do in this case. This indicates that the modified RTR maintains more diversity of the population than FR and ER do. The traditional RTR consumes fewer NFE than the other strategies on the B-traps with $\alpha > 0.65$.

When $\alpha \leq 0.3$, the optimal model does not include any dependency among bits (Fan, Yu, & Lee, 2011). The experiment indicates that ECGA with RTR suffers from spurious dependencies. RTR causes ECGA to consume more NFE than FR and ER do, but the modified RTR resolves this problem. This is consistent with the previous experiment on the OneMax problem.

4 Generalization of the Approach for a General EDA

The one-niche distance metric is shown to reduce the spurious dependencies induced by RTR. Since this approach requires the BB-wise distances between individuals, it cannot be directly incorporated into EDAs which do not generate explicit linkage groups. To generalize this approach, this section proposes a method based on DSM to estimate the one-niche distance between two individuals.

4.1 Differential Mutual Complement

Fan et al. (Fan, Lee, Yu, & Ho, 2010) proposed the differential mutual complement (DMC) to detect pairwise dependencies. Let $H_{x_i=a,x_j=b}$ denote the schema where the $i$-th bit is $a$ and the $j$-th bit is $b$. For simplicity, abbreviate $H_{x_i=a,x_j=b}$ to $H_{ab}$. The definition of DMC can be expressed as follows.

**Definition 3** For a population of binary strings of length $\ell \geq 2$, the DMC between the $i$-th and $j$-th bits ($i \neq j$) is defined as

$$\delta_{ij} = \begin{cases} p_{00}p_{11} - p_{01}p_{10} & \text{if } p_{00} \text{ or } p_{11} \text{ is the largest among } p_{00}, p_{01}, p_{10} \text{ and } p_{11} \\ p_{01}p_{10} - p_{00}p_{11} & \text{otherwise}, \end{cases}$$

(10)

where $p_{ab}$ is the proportion of binary strings containing schema $H_{ab}$ in the population.
Assume that $H_{11}$ is the optimal schema. In the procedure of generating offspring, $H_{11}$ is disrupted if it is crossed with its complement $H_{00}$; it is reproduced when $H_{01}$ is crossed with $H_{10}$. Let $p_{11}$ be the proportion of $H_{11}$ in the mating pool after selection. If the exchange probability is $\frac{1}{2}$, the proportion of $H_{11}$ after the crossover operation, or the model building and sampling procedures in EDAs, can be expressed as

$$p'_{11} = p_{11} - \frac{1}{2}p_{00}p_{11} + \frac{1}{2}p_{01}p_{10}. \quad (11)$$

To ensure the growth of $p_{11}$, $H_{11}$ needs to be protected if the value of $(p_{00}p_{11} - p_{01}p_{10})$ is large enough. This is the reason for the definition of DMC. Individuals containing the optimal schema are assumed to occupy the largest proportion of the population.

DMC was originally designed to detect pairwise dependency for model building. Here it is utilized to estimate whether or not two individuals differ in only one BB. DMC is chosen since it is shown (Fan, Lee, Yu, & Ho, 2010) to be less sensitive to the threshold value than other metrics such as mutual information.

### 4.2 Threshold for DMC

If the DMC between two bits is greater than a certain threshold, the two bits are considered to be dependent, that is, in the same BB. The following describes how we choose the threshold for the DMC between the $i$-th and $j$-th bits.

The proportion of the optimal schema needs to be increased over generations. The selection operator is the main source to increase the proportion. The model building and sampling procedures also affect the proportion. When the $i$-th and $j$-th bits are linked together, let $\Delta p$ denote the change in the proportion between the original population and the offspring. If the two bits are not grouped, the increment of the proportion is $\Delta p = \frac{1}{2} \delta_{ij}$, given the assumption that the exchange probability is $\frac{1}{2}$.

When $\delta_{ij}$ is large enough, the two bits are grouped to minimize the time required for the optimal schema to occupy an enough proportion of the population. However, increasing the average size of BBs raise the population-sizing requirements (Harik, Cantú-Paz, Goldberg, & Miller, 1997; Goldberg, Sastry, & Latoza, 2001; Yu, Sastry, Goldberg, & Pelikan, 2007). The average size of BBs could be increased by one at the upper bound after the two bits are grouped. The required population size grows by two times in this worst case, according to the gambler's ruin model (Harik, Cantú-Paz, Goldberg, & Miller, 1997). Taking into account both the time and population-sizing requirements, the bits are expected to be grouped when

$$2 \left( \frac{1}{\Delta p} \right) \leq \frac{1}{\Delta p} - \frac{1}{2} \delta_{ij}. \quad (12)$$

This inequality yields a conservative condition for grouping the two bits, $\delta_{ij} \geq \Delta p$.

### 4.3 Distance Estimation Based on DSM

With the DMC metric and the threshold, we propose a method based on DSM to estimate whether two individuals are in the same BB. The modified RTR with this estimation method is thus called DSMRTR. A DSM is a matrix that contains the information of pairwise dependencies between every pair of the variables in a problem (Steward, 1981). It also enables users to visualize and analyze the dependencies.

To represent the pairwise dependencies between every pair of bits in an $\ell$-bit problem, the proposed estimation method first constructs a DSM as $M = [m_{ij}]_{\ell \times \ell}$, and the entries are defined as

$$m_{ij} = \begin{cases} 0 & \text{if } \delta_{ij} < \theta_{ij} \\ 1 & \text{otherwise,} \end{cases} \quad (13)$$

where $\theta_{ij}$ is a threshold that is calculated as follows.
Algorithm 2: FillDSM(X, Y, M)

Input:
The original population \( X = \{ \vec{x}_0, \vec{x}_1, \ldots, \vec{x}_{n-1} \} \);
The offspring population, \( Y = \{ \vec{y}_0, \vec{y}_1, \ldots, \vec{y}_{n-1} \} \);
The DSM, \( M = [m_{ij}]_{\ell \times \ell} \);

1. for each \( i, j \in \{1, 2, \ldots, \ell\} \) such that \( i < j \) do
2. Calculate the DMC between the \( i \)-th and \( j \)-th bits, \( \delta_{ij} \), as in Equation 10;
3. Let \( \theta_{ij} \) be the threshold for the DMC value as in Equation 14;
4. if \( \delta_{ij} < \theta_{ij} \) then
5. \( m_{ij} \leftarrow 0 \);
6. else
7. \( m_{ij} \leftarrow 1 \);
8. return;

Algorithm 3: EstimatedOneNicheDist(\( \vec{x}, \vec{y}, M \))

Input:
Individual \( \vec{x} = x_1x_2\cdots x_\ell \);
Individual \( \vec{y} = y_1y_2\cdots y_\ell \);
The DSM, \( M = [m_{ij}]_{\ell \times \ell} \);

Output:
The estimated one-niche distance, \( d \);

1. \( D \leftarrow \{ k \in \ell \mid x_k \neq y_k \} \), where \( \ell = \{1, 2, \ldots, \ell\} \);
2. if \( |D| < 2 \) then
3. \( d \leftarrow |D| \);
4. else if \( \sum_{i,j \in D; i<j} m_{ij} > \frac{1}{4} |D| (|D| - 1) \) then
5. \( d \leftarrow 1 \);
6. else
7. \( d \leftarrow \infty \);
8. return \( d \);

Let \( p_{ab} \) and \( q_{ab} \) denote the proportions of \( H_{ab} \) in the offspring population and in the original population, respectively. From Section 4.2, the threshold for DMC is chosen to be

\[
\theta_{ij} = p_{uv} - q_{uv},
\]

where \( p_{uv} \) is the largest among \( p_{00}, p_{01}, p_{10} \) and \( p_{11} \). The process of filling the DSM is presented as Algorithm 2.

A subset of individuals in the original population is chosen for an offspring individual \( \vec{y} \). For an individual \( \vec{x} \) in this subset, let \( D = \{ d_1, d_2, \ldots, d_{|D|}\} \subset \{1, 2, \ldots, \ell\} \) be a set containing the indexes of the bits in which \( \vec{x} \) and \( \vec{y} \) differ. If \( D = \emptyset \) or \( |D| = 1 \), the BB-wise distance between \( \vec{x} \) and \( \vec{y} \) is evidently zero or one, respectively. Otherwise, the BB-wise distance is estimated to be one if

\[
\sum_{i,j \in D; i<j} m_{ij} \geq \frac{1}{4} |D| (|D| - 1).
\]

The summation represents how many pairs of the bits in which \( \vec{x} \) and \( \vec{y} \) differ are dependent. If the summation is greater than half the number of pairs in \( D \), all the bits are estimated to be in the same BB. The estimated one-niche distance can be calculated as shown in Algorithm 3 and the whole process of DSMRTR is presented as Algorithm 4.

To examine the estimation of the one-niche distances, we compare DSMRTR and the modified RTR on the B-trap problem. The experiment as shown in Figure 4 indicates that DSMRTR performs nearly identically to the modified RTR for ECGA.
Algorithm 4: DSMRTR

Input:
The original population, \( X = \{ \vec{x}_0, \vec{x}_1, \ldots, \vec{x}_{n-1} \} \);
The offspring population, \( Y = \{ \vec{y}_0, \vec{y}_1, \ldots, \vec{y}_{n-1} \} \);
The window size, \( w \) (\( w \leq n \));

Output:
The population \( X \) for the next generation;

1. Construct a matrix \( M = [m_{ij}]_{\ell \times \ell} \);
2. FillDSM(\( X, Y, M \));
3. for each \( \vec{y} \in Y \) do
   4. Choose a random subset \( S \subset N \) consisting of \( w \) integers, where \( N = \{0, 1, \ldots, n-1\} \);
   5. Let \( m \) be any element in \( \text{argmin}_{i \in S} \text{EstimatedOneNicheDist}(\vec{x}_i, \vec{y}, M) \);
   6. if \( \text{Fitness}(\vec{y}) > \text{Fitness}(\vec{x}_m) \) then
      7. \( \vec{x}_m \leftarrow \vec{y} \);
8. return \( X \);

Figure 4: NFE required for ECGA to solve the 100-bit problem composed of 20 concatenated B-traps of order 5 (\( \ell = 100 \)) with different values of \( \alpha \). DSMRTR is compared with the modified RTR to examine the estimation method.

Filling the DSM requires time of \( O(n\ell^2) \). Estimating the one-niche distance between two individuals requires time of \( O(\ell^2) \). The estimation is performed for \( w \cdot n \) times in a replacement procedure. The time complexity of DSMRTR is thus \( O(wn\ell^2) \). This overhead does not grow faster with the problem size than that of a general model-building process, such as \( O(n\ell^3) \) for ECGA (Harik, Lobo, & Sastry, 2006). Moreover, many model-building techniques require counting the occurrences of pairs of variables in the population. By using more storage space, the results of the counting can be shared with DSMRTR, and the overall computational cost can be reduced.

5 Experiments and Discussion

In this section, we incorporate DSMRTR into hBOA. Experiments on traps (Deb & Goldberg, 1993) and the OneMax problem are conducted to verify whether or not DSMRTR reduces spurious dependencies and maintains enough diversity of the population. Discussion on the results is then presented.

5.1 DSMRTR and Spurious Dependencies

BOA/hBOA uses Bayesian networks to model the dependencies among the variables of a problem. The quality of a network structure is quantified by scoring metrics such as the Bayesian information criterion (BIC) (Schwarz, 1978) or the K2 metric (Chickering, Heckerman, & Meek, 1997). The
conditional probabilities for each variable given all possible instances of the parent variables can be stored in the form of local structures such as decision trees or decision graphs.

Lima et al. (Lima, Pelikan, Goldberg, Lobo, Sastry, & Hauschild, 2007) found that RTR caused BOA to generate more spurious dependencies on the $m$-$k$ trap problem. In their work, Bayesian networks with decision trees and the K2 metric were used. The fitness function of the $m$-$k$ trap problem can be expressed as

$$f_{\text{trap}}(\vec{x}) = m - 1 \sum_{i=0}^{t_{k}(\vec{x})} x_{ik+1} x_{ik+2} \cdots x_{ik+k}, \quad (16)$$

where

$$t_{k}(\vec{x}) = \begin{cases} 1 & \text{if } u(\vec{x}) = k \\ k - 1 - u(\vec{x}) & \text{otherwise.} \end{cases} \quad (17)$$

We experiment on the trap problem with hBOA which utilizes Bayesian networks with decision graphs and the BIC metric. Let $X_i$ denote a random variable whose observation is $x_i$. The decision graph encodes the random variables as nodes and the dependencies between two variables as directed edges. In the $m$-$k$ trap problem, a variable of each trap subfunction should only depend on some of the remaining $k - 1$ variables. For example, edge $\{X_2, X_1\}$ represents a correct linkage, while edge $\{X_7, X_1\}$ represents a spurious linkage.

At each generation, the number of spurious linkages in the decision graph is measured. Considering that hBOA with ER requires the largest population (see Figure 8 (b)), the population sizes are set to be the minimum required number for hBOA with ER to perform a successful run. A run is considered successful when a solution consisting of optimal partial solutions in all but one trap partition is found.

With the same population size, Figure 5 shows that RTR induces more spurious linkages than other replacement strategies do. This verifies that the problem of spurious dependencies also holds for hBOA with decision graphs and the BIC metric. DSMRTR is shown to reduce the number of spurious linkages.

We also conduct an experiment on the OneMax problem. Similar methodologies are used, except that a run is considered successful when a solution different from the global optimum in only one bit is found. In this problem, any edge in the decision graph represents a spurious linkage. Figure 6 shows that DSMRTR induces fewer spurious linkages on the OneMax problem than RTR does.

### 5.2 Niching Capabilities of DSMRTR

We also use the 100-bit concatenated trap problem to test the niching capabilities. This problem consists of 20 BBs and 2 optimal peaks within each BB. The necessary number of niches to solve this problem is thus $2 \times 20 = 40$ as described in Section 5.1.
Figure 6: The number of spurious linkages in the decision graph of hBOA over generations, on the 100-bit OneMax problem.

Figure 7: The number of local optima maintained by hBOA over generations, on the problem composed of 20 concatenated traps of order 5 ($\ell = 100$).

The number of different local optima maintained in the population is measured over the generations. Figure 7 presents the results. FR and ER do not preserve multiple local optima after a certain number of generations. The number of local optima preserved by DSMRTR also decreases over generation, but the decline is much less substantial. DSMRTR still preserves more than the necessary number of niches over tens of generations.

Figure 8(a) presents the NFE required for hBOA on the trap problem with different lengths. While DSMRTR has been shown to induce fewer spurious dependencies than RTR does, the NFE requirements of DSMRTR is near to that of RTR. Figure 8(b) shows that DSMRTR requires a smaller population than FR and ER do but a large population than RTR does.

5.3 Population-Sizing for DSMRTR

Given the assumptions described in Section 3.1, ideally the modified RTR or DSMRTR is equivalent to RTR with respect to preserving the necessary niches on a decomposable problem. However, the results shown in Figures 3 and 5 point out that RTR still consumes slightly fewer NFE or requires a smaller population on highly deceptive problems.

One possible reason is that the EDA models do not always correctly describe the problem structure. If the models are inaccurate, the one-niche distance measure does not follow the intent of determining whether or not two individuals occupy the same niche. In DSMRTR, most pairs of individuals are regarded to be in different niches in the initial generations, when most BBs have not been learned by EDAs.

In this case, an individual is almost randomly chosen to compete with an offspring individual, and DSMRTR effectively does not perform niching. EDAs with DSMRTR thus require a larger population on highly deceptive problems than those with RTR, in order to ensure the correct detection of important dependencies in the initial generations.
Figure 8: The required NFE and population size for hBOA to solve the concatenated trap problem of order 5 with different problem sizes.

6 Conclusion

This paper examined the effects of RTR on the quantity of spurious dependencies and confirmed that excessively distorting the distribution of individuals is a cause. The one-niche distance metric was defined to reduce the distortion. RTR with this metric was shown to induce fewer spurious dependencies for ECGA, which generates explicit linkage groups. To generalize the approach for EDAs which do not provide explicit linkage information, this paper proposed DSMRTR. It constructed the DSM via DMC to estimate one-niche distances and was also shown to induce fewer spurious dependencies for hBOA.

Several challenges remain in this work. First, DSMRTR requires a larger population on highly deceptive problems, in order to ensure the correct detection of important dependencies. Possible solutions to this problem include revising the design of the one-niche distance metric, refining the estimation of distances, or combining RTR and DSMRTR to create a two-phase niching scheme. Furthermore, the behavior of DSMRTR on problems with hierarchical or overlapping BBs needs to be investigated. Since this paper focuses only on the case where the binary tournament selection is used, the choice of selection methods can be also considered.

This paper presents a preliminary research on balancing both the diversity of the population and the accuracy of EDA models. Existing designs of EDAs have focused on the model-building techniques. However, commonly used operators in EDAs such as the RTR and tournament selection were designed for traditional genetic algorithms and did not consider model building. This paper shows that a replacement strategy which takes EDA models into account benefits model building. It is thus promising to further refine the approaches in this paper or to develop different selection strategies. In the future, replacement and selection strategies which preserve the model accuracy could further enhance the performances of EDAs.
References


