Effective Structure Learning in Bayesian Network Based EDAs

S. Ivvan Valdez, Arturo Hernández, and Salvador Botello

Centro de Investigación en Matemáticas A.C.
C.P. 36240, Guanajuato, Guanajuato, Mex.
{ivvan,artha,botello}@cimat.mx

Abstract. Estimation of Distribution Algorithms (EDAs) is a high impact area in evolutionary computation and global optimization. One of the main EDAs strengths is the explicit codification of variable dependencies. The search engine is a joint probability distribution (the search distribution), which is usually computed by fitting the best solutions in the current population. Even though using the best known solutions for biasing the search is a common rule in evolutionary computation, it is worth to notice that most evolutionary algorithms (EAs) derive the new population directly from the selected set, while EDAs do not. Hence, a different bias can be introduced for EDAs. In this article we introduce the so called Empirical Selection Distribution for biasing the search of an EDA based on a Bayesian Network. Bayesian networks based EDAs had shown impressive results for solving deceptive problems, by estimating the adequate structure (dependencies) and parameters (conditional probabilities) needed to tackle the optimum. In this work we show that a Bayesian Network based EDA (BN-EDA) can be enhanced by using the empirical selection distribution instead of the standard selection method. We introduce weighted estimators for the K2 metric which is capable of detecting better the variable correlations than the original BN-EDA, in addition, we introduce formulas to compute the conditional probabilities (local probability distributions). By providing evidence and performing statistical comparisons, we show that the enhanced version: 1) detects more true variable correlations, 2) has a greater probability of finding the optimum, and 3) requires less number of evaluations and/or population size than the original BN-EDA to reach the optimum. Our results suggest that the Empirical Selection Distribution provides to the algorithm more useful information than the usual selection step.

Keywords: Estimation of Distribution Algorithms, Selection Methods Selection Distribution, Empirical Selection Distribution.

1 Introduction

Estimation of Distribution Algorithms (EDAs) are a family of global optimization algorithms which main strengths are related with: the explicit codification of variable dependencies and the use of self-learned parameters for performing

O. Schütze et al. (eds.), *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III*, Studies in Computational Intelligence 500,
DOI: 10.1007/978-3-319-01460-9_1,

the optimum search. EDAs were derived from probabilistic modeling of genetic algorithms.

In evolutionary computation literature it is well known that the simple genetic algorithm (GA), and most of the standard GA approaches, suffer negative effects of the building block disruption when the decision variables are highly correlated. That is to say, several specific variable instances must be generated or preserved in order to increase the probability of sampling the optimum. The problem was called: the learning linkage problem in the context of evolutionary algorithms [5], and is one of the main motivations of evolutionary computation researchers to propose new methods and frameworks such as Estimation of Distribution Algorithms (EDAs) [14] [17] [24].

Even though the firsts EDAs approaches consider independent variables [14] [6] [2], soon after graphical models were used to improve the search by integrating information about variable correlations [3] [18]. bivariate EDAs shown that they can efficiently solve problems that the simple genetic algorithm and univariate EDAs can not. Regarding the encouraging results, researchers then propose to use more complex models than bivariate, resulting in more powerful algorithms. One of the first algorithms intended to exploit high order correlations was the Factorized Distribution Algorithm [1]. The FDA uses a factorization of the search distribution to perform the search, the original approach does not propose a method to infer such factorization or distribution structure, then other works extended the original to integrate structure-learning methods[21].

One of these approaches is the Bayesian Optimization Algorithm (BN-EDA) [17,11,12]. It is a powerful EDA to solve deceptive-like problems, and in general decomposable or nearly decomposable problems. A general Bayesian Network based algorithm (BN-EDA) is presented in Algorithm 1, [11]. In Step 3 the selection step could use different selection methods to select a subset of the most promising solutions of the current population. In Step 4 the structure and parameters of a Bayesian network are learned from the selected set. The enhancement of these two steps is the main contribution of this article.

Algorithm 1. Bayesian Network based estimation of Distribution Algorithm

1 Create a random population $\mathbb{X}^t$ of n_{pop} individuals;
2 Evaluate the population $\mathbb{X}^t$, $\mathbb{F} \leftarrow f(\mathbb{X}^t)$;
3 Select n_{sel} individuals from $\mathbb{X}^t$ using a selection procedure $\mathbb{S} \leftarrow selection(\mathbb{X}^t, \mathbb{F})$;
4 Model $\mathbb{S}$ by learning the most adequate Bayesian network B;
5 Create a new population $\mathbb{X}^{t+1}$ by sampling from the joint probability distribution of B;
6 Evaluate population $\mathbb{X}^{t+1}$;
7 Replace all (or some) individuals in population $\mathbb{X}^t$ by those from $\mathbb{X}^{t+1}$;
8 If stopping criteria are not satisfied, return to step 3.

Besides the ability of solving hard optimization problems which require the use of variable dependencies, the BN-EDA provides of additional interesting features, such as:

- It intends to discover an adequate structure of the problem, in order to use it for sampling new high-quality candidate solutions. Considering that the knowledge of the problem structure can be as valuable as the optimum approximation [11], this is a remarkable feature of the BN-EDA.
- There exist model-efficiency techniques which take advantage of the explicit structure codification which leads to important speed ups. For instance, using previous models obtained from several runs to tackle a similar optimization problem [7]. Or, an evaluation relaxation technique [13], which uses an entropy based measure to decide if a candidate solution must be evaluated.
- A priori knowledge could be integrated in the structure and parameter learning procedure.
- For a theoretical Bayesian network based Algorithm called the Factorized Distribution Algorithm (FDA) [16], it has been shown that linear scaling of the population size with n_{var} (number of variables) is sufficient to converge to the optimum, even for hard optimization problems [15].

Nevertheless our article is focus in provide evidence about the enhancements of the empirical selection distribution in the original BN-EDA, the reader must notice that all the improvements and features just listed are shared or can be applied in the original BN-EDA as well as in our approach.

In order to introduce the Empirical Selection Distribution, recall that, convergence to the optimum has been proved for theoretical EDAs [24], by using the exact **selection distribution**.

The selection distribution is defined as the underlying distribution of an infinite sized selected set. It has been shown that the selection distribution depends on the objective function [24]. Thus, the greater the objective function of a point is, the greater the probability associated to such point have to be, in consequence, for the next generation the most fitted individuals are intensively sampled. The selection distribution is defined for infinite sized populations and it can not be directly used in practical approaches.

The **empirical selection distribution** [23] is derived from the theoretical model of the exact selection distribution, but considering a finite sized population. It can be considered as an *a priori* probability or relative frequency for each individual in the population, and then, can be used to estimate the search distribution parameters. Hence, it unifies the selection and estimation steps resulting in a new procedure for biasing the search.

This article introduces a BN-EDA algorithm which uses the empirical selection distribution for estimating the structure as well as the conditional probabilities. According to Mühlenbein [15], a BN-EDA algorithm converges to the optimum if sufficient variable correlations are learned, and there is a large enough probability of sampling the optimum. In this vein, we show, by using the most widely

reported objective functions in BN-EDA (by instance: Deceptive-3 and Trap-5), that the approach introduced in this article finds more true variable dependencies than the original BN-EDA, and the probability of finding the optimum is also greater than the original. Additionally, our results show that our approach requires a smaller population size than the original BN-EDA, which impacts on the computational cost for solving hard optimization problems. All the promissory results in this article can be explained as an effect of the empirical selection distribution, considering that it is the unique extra feature we add to the original BN-EDA.

The paper is presented as follows: Section 2 briefly reviews the most common selection methods used in EDAs. Section 3 describes the procedure to integrate the empirical selection distribution for computing the structure and parameters of a BN-EDA EDA. A set of experiments to contrast the empirical selection distribution based BN-EDA(ESD-BN-EDA) with the original BN-EDA is presented in Section 4. Finally, Section 5 presents the main conclusions.

2 Selection Methods

The main goal of a selection operator is to bias the population towards promising regions of the search space. The most common selection methods: truncation, Boltzmann, proportional and tournament are a kind of *subset selection* methods. This kind of method selects a fraction of the population, then some individuals are represented and some others are not. Subset selection can be seen as a weighting method (for the parameter computation) which associates a weight proportional to the number of times an individual is selected, and 0 otherwise. In the case of the truncation method the weights become binary. In the case of the other selection methods, they use a random process to sample the selected set from the population, it is possible (with an small probability, but possible) that even the best solution is not represented in the selected set. Hence, practical approaches which use the subset selection does not maintain a direct relationship between the objective function and representation intensity in the parameter computation, and by consequence neither between the objective function and the posterior search distribution. The possible undesired effects of this lost are:

- Solutions with a high objective value could not be represented or could be misrepresented in the search distribution, because they could not be selected, or their frequencies in the selected set does not correspond with the objective value.
- The selected set very often covers a smaller region than the population, or represents less instances (in discrete space) than the population. Thus a natural variance reduction is expected due to the subset selection [22].
- Information about promising regions could be lost even if there are solutions in the population that indicates such regions, due to the random selection process in most of the subset selection methods.

– Due to the fact that different selected sets could be obtained, by the subset selection method, if it is applied several times over the same population, the performance of the algorithm is not so confident, thus we expect more variance in the performance than methods which always compute the same search distribution from the same population.

On the other hand, the theoretical selection distributions directly depend on the objective function, by instance the proportional selection distribution can be written as Equation 1.

$$p(x_i) = \frac{f(x_i)}{\sum_{j=1}^{n_{instances}} f(x_j)} \tag{1}$$

Where x_i is a possible instance of the decision variables, $f(x_i)$ is the objective function value of x_i, and $n_{instances}$ is the number of possible instances. Note that the exact selection distribution requires to know the objective values of all possible instances.

The empirical selection distribution (ESD) [23] is the counterpart model of the exact selection distribution, when considering a finite population. It intends to explicitly relate the objective function value with the search distribution.

The ESD for the most common selection methods is shown in Table 1. S_t is the selected set, and $p(x_i)$ is a probability associated with the individual i. The ESD has been used in a continuous EDA with a Gaussian distribution [23]. In this article we introduce a Bayesian network (BN) based EDA which uses the ESD. The $p(x_i)$ are used to estimate the structure and parameters of the BN as it is shown in the next section. There are some remarkable features of the ESD:

– It considers the whole population to be computed. Consequently we are using all the available information in the population (notice that subset selection throws away an important fraction of the population). Meaning that all the points in the population are represented.
– Considering that all the population is used, hence, a wider region is covered than using a subset of the population, in consequence we expect a wider variance if needed (if high-fitness individuals are spread over the search space), or a reduced variance (if the high-fitness points are in the same reduced region).
– The sampling intensity correspond to the fitness value.

The next section shows how to integrate the ESD in the structure and parameter computation of a Bayesian network.

3 Estimating the Structure and Parameters of the Bayesian Network

A Bayesian network is a probability model which encode the joint probability distribution for a set of variables. A Bayesian Network for variables

Table 1. Empirical selection distribution model for a finite-sized population

Empirical Selection Distribution
Truncation $\hat{p}^S(x_i,t) = \begin{cases} \frac{1}{\lvert S_t \rvert} \text{ if } f(x_i) \geq \theta_t \\ 0 \quad \text{otherwise} \end{cases}$ $\lvert S_t \rvert = \sharp$ of individuals with $f(x_i) > \theta_t$
Proportional $\hat{p}^S(x_i,t) = \frac{f(x_i)}{\sum_{j=1}^{\lvert X \rvert} f(x_j)}$
Binary Tournament $\hat{p}^S(x_i,t) = \frac{\sum_{j=1}^{\lvert X \rvert} I(i,j)}{\sum_{i=1}^{\lvert X \rvert} \sum_{j=1}^{\lvert X \rvert} I(i,j)}$ Where $I(i,j) = 1$ if $f(x_j) < f(x_i)$ and 0 otherwise

$\mathbf{X} = \{X_1, ..., X_n\}$ consist of: 1) a network structure $\mathbf{S}$ that encodes a set of conditional independence assertions about variables in $\mathbf{X}$, and 2) a set $\mathbf{P}$ of local probability distributions associated with each variable. The network structure $\mathbf{S}$ is a directed acyclic graph. The nodes in $\mathbf{S}$ are in one-to-one correspondence with the variables $\mathbf{X}$ [9]. Given structure $\mathbf{S}$, the joint probability distribution for $\mathbf{X}$ is given by Equation 2

$$p(x) = \prod_{i=1}^{n} p(x_i|\pi_i) \tag{2}$$

The local probability distributions are the corresponding to the terms in the product of Equation 2. Π_i are the parents of the variable X_i.

In order to estimate or learn a Bayesian network from data it is necessary two components: a scoring metric, and a search procedure. The search procedure proposes a candidate Bayesian network, while the scoring metric discriminate among the proposed networks [10].

For this article we use the same search procedure than the original BOA [17], a greedy algorithm with edge addition. The procedure starts with an empty network (without edges), then we test adding a single edge to each variable, the edge that increases the most the scoring metric is actually added to the network. The process is repeated until the metric value can not be increased or the network can not be maintained acyclic.

The scoring metric is modified according to the Empirical Selection Distribution as is shown in the next subsection.

3.1 The K2 Scoring Metric

According to Lima et al. [11] the K2 metric delivers better results for the BOA, in terms of model accuracy, than the BIC metric. Hence, in this article as well as in the original BOA [17] we use the K2 metric for our Bayesian network based algorithm. The K2 metric can be derived from the BDe metric [10] in Equation 3.

$$P(B,S) = P(B)\prod_{i=1}^{n}\prod_{j=1}^{q_i}\left(\frac{\Gamma(N'_{ij})}{\Gamma(N_{ij}+N'_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(N_{ijk}+N'_{ijk})}{\Gamma(N'_{ijk})}\right) \quad (3)$$

Where: r_i are the number of states of the finite random variable X_i. $q_i = \prod_{X_j \in \Pi_i} r_j$ is the number of possible configurations of the parent set Π_i of X_i. w_{ij} is the $j-th$ configuration of the parents Π_i, $(1 \leq j \leq q_i)$. N_{ijk} is the number of instances in the data S (in this case the selected set), where the variable X_i takes its $k-th$ value x_{ik} and the variables in Π_i take their $j-th$ configuration w_{ij}. $N_{ij} = \sum_{j=1}^{r_i} N_{ijk}$ is the number of instances in the data S where the variables in Π_i take their $j-th$ configuration w_{ij}.

The K2 variant of the BDe metric considers no prior knowledge about the instances. Consequently, $N'_{ijk} = 1$, and $N'_{ij} = r_i$, in Equation 4.

$$P(B,S) = P(B)\prod_{i=1}^{n}\prod_{j=1}^{q_i}\left(\frac{r_i}{\Gamma(N_{ij}+r_i)}\prod_{k=1}^{r_i}\frac{\Gamma(N_{ijk}+1)}{\Gamma(1)}\right) \quad (4)$$

Notice that a logarithmic version of the K2 metric can be use in order to reduce computational errors and effort.

3.2 Modifying the K2 Metric with the ESD

In order to use the ESD for computing the K2 scoring metric, we define a *virtual sample size* greater than n_{pop}, $n_{virtual} >> n_{pop}$. The computation of N_{ijk} and N_{ij} is as shown in Algorithm 2. We show both computation algorithms in the same loop, although it can be in a different loop for practical purposes. Notice that the computation is performed by using a floating point value with $\hat{N}_{ij}$ and $\hat{N}_{ijk}$, and rounded at the end of the computation, in order to reduce rounding errors. Using N_{ij} and N_{ijk} computed as shown in Algorithm 2 we can proceed to compute the K2 metric as usual. $p(x_i)$ in Algorithm 2 is the empirical selection distribution value associated with the individual x_i, for $i = 1..n_{pop}$.

Algorithm 2. Computation of N_{ijk} and N_{ij} using the ESD

```
1  N̂_ij = 0;
2  N̂_ijk = 0;
3  for Each individual x in the population 𝕏 do
4  |  if x_i takes the value x_ik and the parents Π_i takes the value w_ij then
5  |  |  N̂_ijk = N̂_ijk + (n_virtual)p(x_i);
6  |  end
7  |  if The parents Π_i takes the value w_ij in x then
8  |  |  N̂_ij = N̂_ij + (n_virtual)p(x_i);
9  |  end
10 end
11 N_ij = integer(N̂_ij + 0.5);
12 N_ijk = integer(N̂_ijk + 0.5);
```

3.3 Computing the Conditional Probabilities

Once we have obtained the Bayesian Network structure by using the search process and the scoring metric, then we proceed to compute the parameters or probabilities to sample by using Equation 5.

$$\hat{p}_{ijk} = \hat{P}(X_i = x_{ik} | \Pi_i = w_{ij}) = N_{ijk}/N_{ij} \tag{5}$$

In the case of the ESD-BN-EDA, N_{ij} and N_{ijk} are computed as shown in Algorithm 2. Due to rounding errors, it is possible that a normalization procedure be needed in order to ensure that the parameter is actually a probability (that it sums 1). Then we apply the Equation 6, the sum is for each instance x_{ijk} used to compute the probabilities associated with the variable X_i.

$$p_{ijk} = \frac{\hat{p}_{ijk}}{\sum_{x_{ijk}} \hat{p}_{ijk}} \tag{6}$$

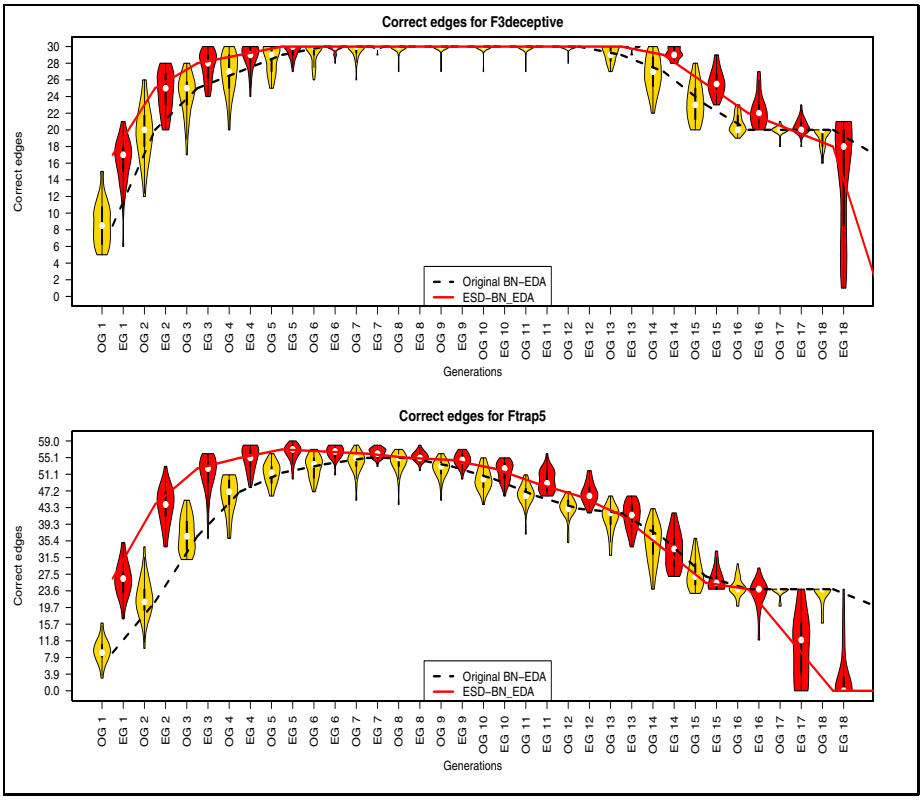

Fig. 1. Comparison of the correct edges added when using the binary tournament

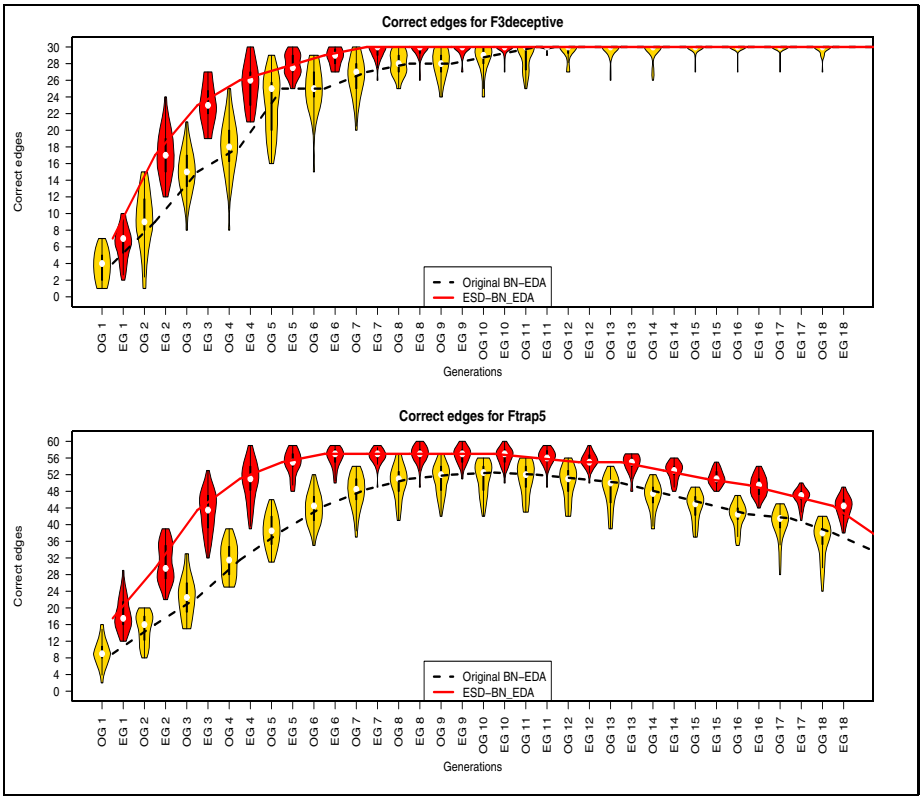

Fig. 2. Comparison of the correct edges added when using the proportional selection

4 Experiments and Performance Analysis

The experiments in this Section are intended to provide evidence about the boosted performance of the BN-EDA when it uses the Empirical Selection Distribution. The information we desire to determine from the experiments is related with three main topics:

1) If the ESD is helpful to determine better the truth structure of the problem. 2) If the ESD increases the probability of finding the optimum, and 3) If the ESD has a beneficial impact on the number of evaluations and population size needed to find the optimum.

The comparisons are performed by using the problems in the original BOA [17], as well as in other research paper which investigate the BOA performance under different selection conditions [11] [12]. The test problems are defined in Table 4

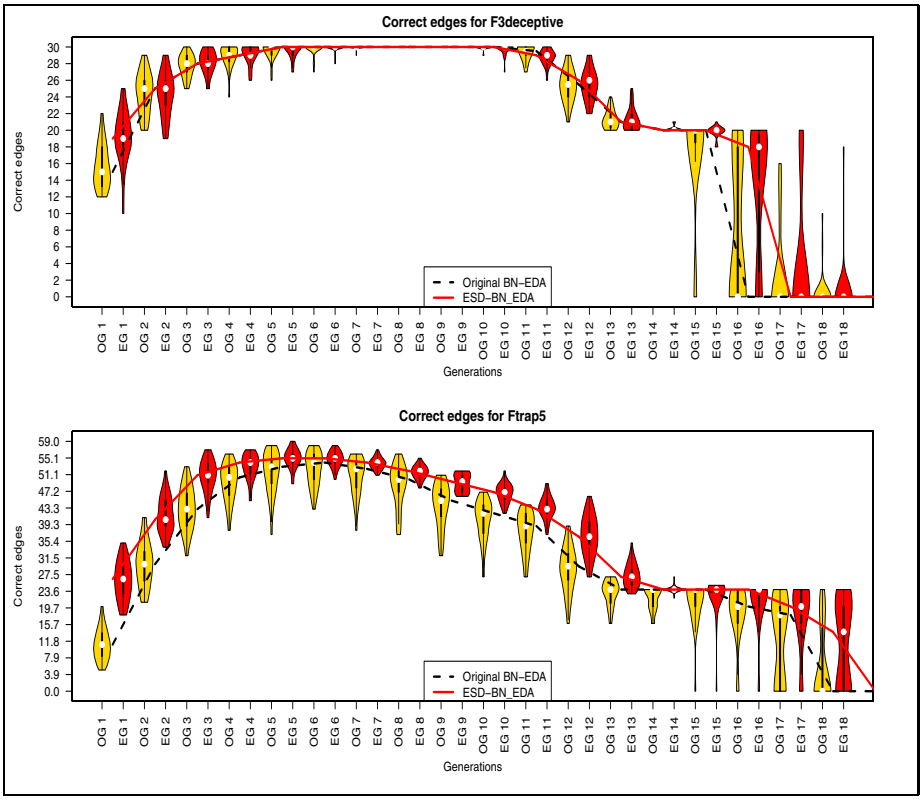

Fig. 3. Comparison of the correct edges added when using the truncation selection

4.1 Evidencing the Number of Correct Dependencies Captured

Experiment Description. For the objective functions presented in Table 4 we define a random order I_i for the variables x_i. This order is used to define the correlations. For example, suppose that we define the order $I = [3, 9, 4, 6, 10, 1, 8, 5, 2, 7]$ for the f_{trap5}, then, the objective function is called using $f_{trap5}(x_3, x_9, x_4, x_6, x_{10})$ and $f_{trap5}(x_1, x_8, x_5, x_2, x_7)$. Hence we expect to add edges in BN which relate the variables in each one of the sets. Under this order we known that the related variables are $[3, 9, 4, 6, 10]$ and $[1, 8, 5, 2, 7]$. We count how many edges among related variables are added each generation (correct edges). We contrast the correct edges added by the enhanced ESD-BN-EDA versus the original. Using violin plots we graphically show the differences in variance, median, and density (or empirical distribution of the edges) during the generations. Additionally, we perform hypothesis tests to known if the difference between correct edges of ESD-BN-EDA and the original are statistically significant.

Additively composed function: $f(x) = \sum_{i=0}^{l-1} f_k(u)$. where $u = \sum_{j \in S_i} x_j$, and S_i is a partition of 3 or 5elements of the set $\{1..n_{var}\}$, n_{var} is the number of decision variables, and f_k is one of the following:	
Deceptive order 3	$f_{3deceptive} \begin{cases} 0.9 & if\ u = 0 \\ 0.8 & if\ u = 1 \\ 0 & if\ u = 2 \\ 1 & if\ otherwise \end{cases}$
Trap order 5	$f_{trap5} \begin{cases} 4 - u & if\ u < 5 \\ 5 & otherwise \end{cases}$

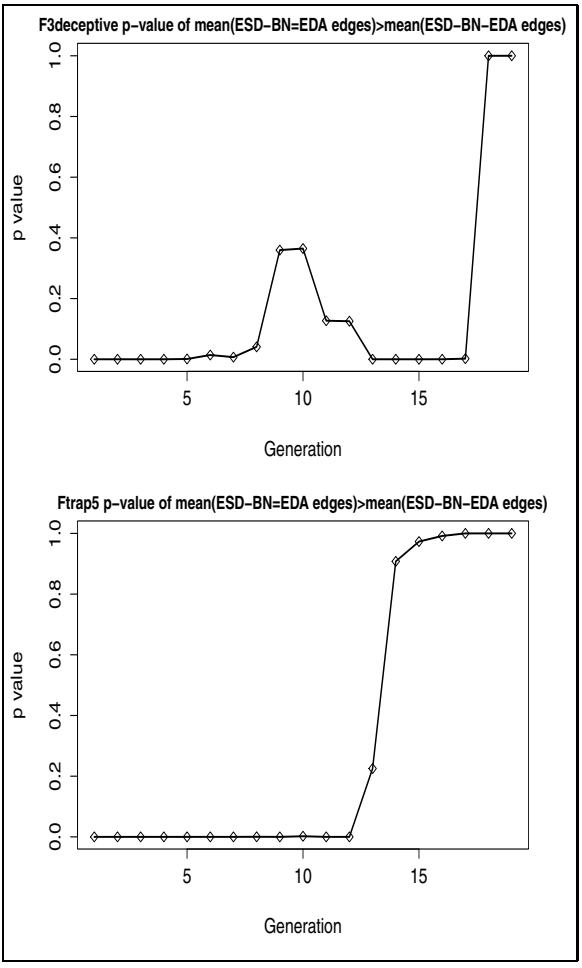

Fig. 4. $p-value$ from hypothesis test about the number of correct edges for the binary tournament

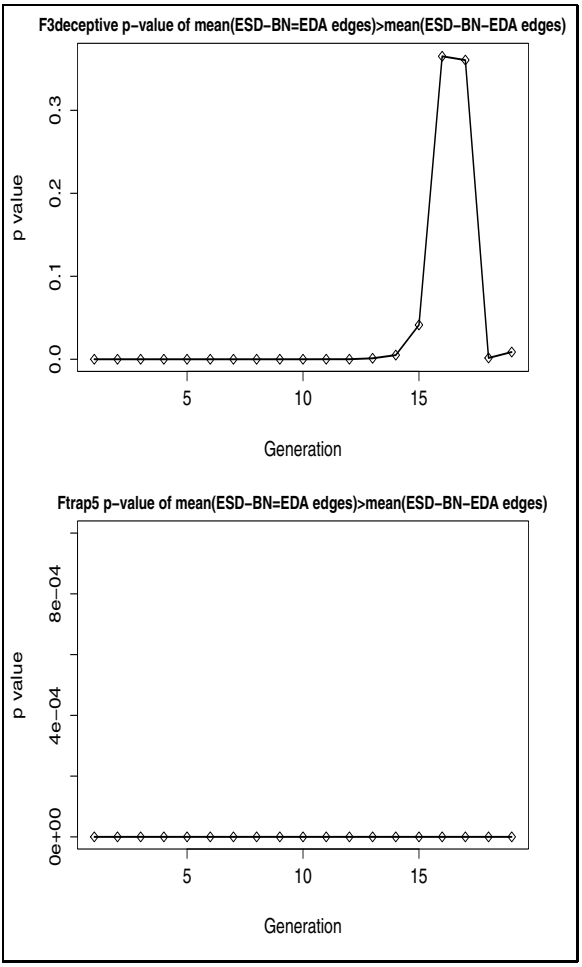

Fig. 5. $p - value$ from hypothesis test about the number of correct edges for the proportional selection

Experiment Settings. We perform 30 independent runs for the test problems in Table 4. The number of variables is 30. We use the first 20 generations for the comparison because this number is enough for convergence of the algorithm. According to [11], the firsts generations is when most of the correct edges are detected, when the convergence is found adding an edge poorly increases the score, and the model is over-fitted adding spurious edges. The population sizes are $\{900, 1300\}$ and the maximum number of allowed parents is $k = \{2, 4\}$ as suggested by Pelikan et al. [17], for the $f_{3deceptive}$ and the f_{trap5} respectively.

Experimental Results. In Figures 1, 2 and 3, we show violin plots of the *correct edges* discovered by each of the algorithms. The violin plots are similar to box plots but they give a better look of the data distribution: the central

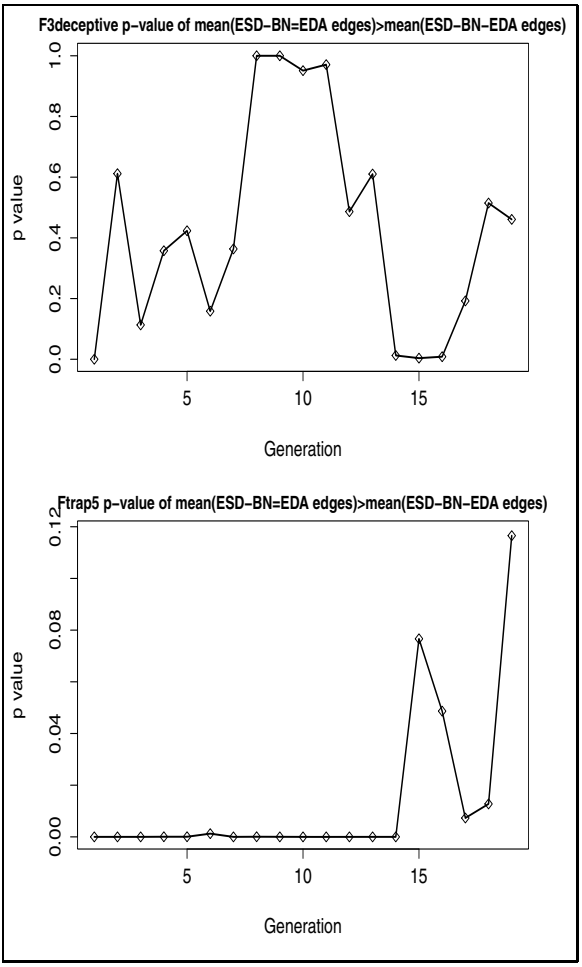

Fig. 6. $p - value$ from hypothesis test about the number of correct edges for the truncation selection

dot is the median of the data, the length of the violin plot measured along the $y - axis$ shows the data dispersion, while the shape of the "box" is a kernel-density approximation to the data density. Hence, the largest plots means the highest variance of the data, in this case the largest violin plots means highest variance in the number of correct edges discovered each generation. The violin plots labeled as "OG x" are computed with data from the x generation of the original BN-EDA, while the violin plots labeled with "EG x" is the corresponding x generation of the Empirical Selection based BN-EDA (ESD-BN-EDA). In Figures 1 to 3, the higher violin plots (with the central dot higher in the $y-axis$) represent the generations and algorithm which discovers more edges. In order to compare more precisely which algorithm discovers more edges, we draw a line

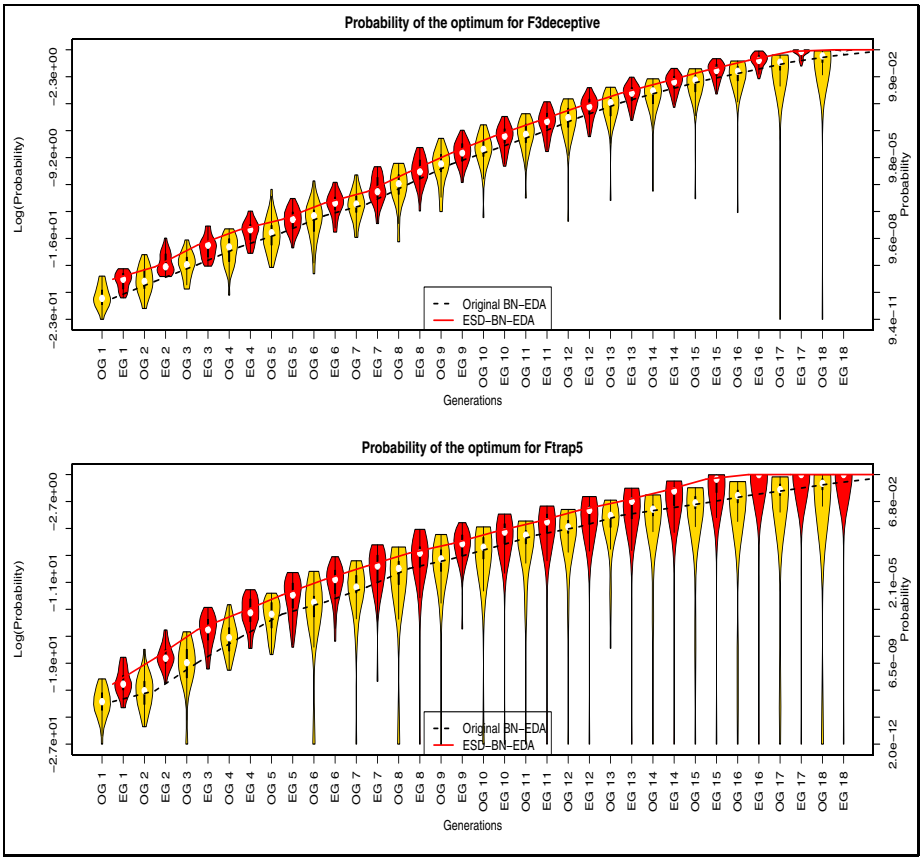

Fig. 7. Probability of sampling the optimum for the binary tournament

with the median of the correct edges discovered, with the same coordinates in the $x - axis$, and the corresponding median in the $y - axis$: the dashed line is the original BN-EDA while the solid line corresponds to the ESD-BN-EDA. Additionally, violin plots let us know which algorithm is the most robust, because more mass around the center as well as a smaller violin, indicate a more consistent performance. For this comparison, another interesting characteristic is that in most of the cases the violin plots show a single mode, which means that most of the times the number of discovered edges is similar for the same generation.

Special Note. When using truncation selection, we perform almost the same estimation for the original BN-EDA and the ESD-BN-EDA, the only difference is the virtual sample and the general procedure of the estimation (for example the calculus of log functions with greater numbers), but the same input information is used to learn the BN. So, we expect a similar behavior of both algorithms

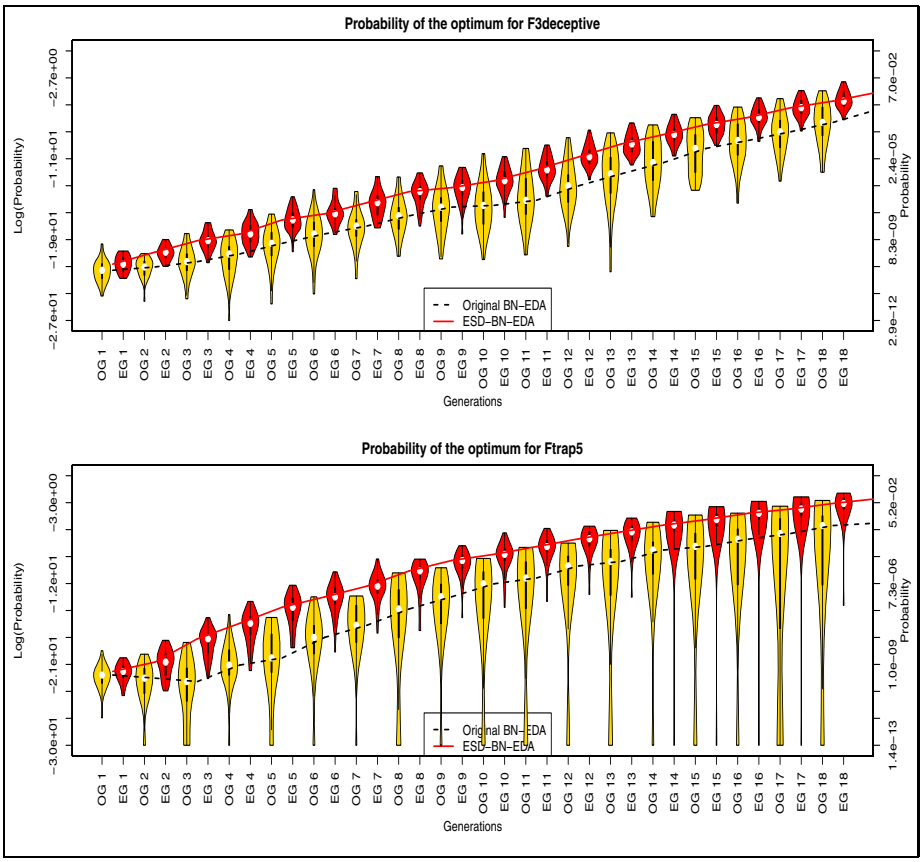

Fig. 8. Probability of sampling the optimum for the proportional selection

for this selection method, and as expected we obtain similar result as they are shown in Figure 3.

The Figures 4, 5 and 6 show the *p-value* from hypothesis tests which compare the means of the number of correct edges discovered each generation by the original BN-EDA versus the ESD-BN-EDA. The hypothesis test is performed by using the Boostrap methodology [4] with 20000 re-samples per generation. In Figure 4 we show the p-value for the mean of correct edges using binary tournament selection. When the p-value is approximately 0, it means that there is strong evidence to say that the ESD-BN-EDA captures more correct edges than the original BN-EDA. Take into account that a 1 value in the p-value does not means that the original BN-EDA captures more correct edges than our approach. A 1 p-value indicates that there is not strong evidence to say the ESD-BN-EDA captures more correct edges. According to Figure 1, we can observe that the ESD-BN-EDA *always* captures at least the same number of correct edges than the original BN-EDA. The 1 p-value in Figure 4 correspond

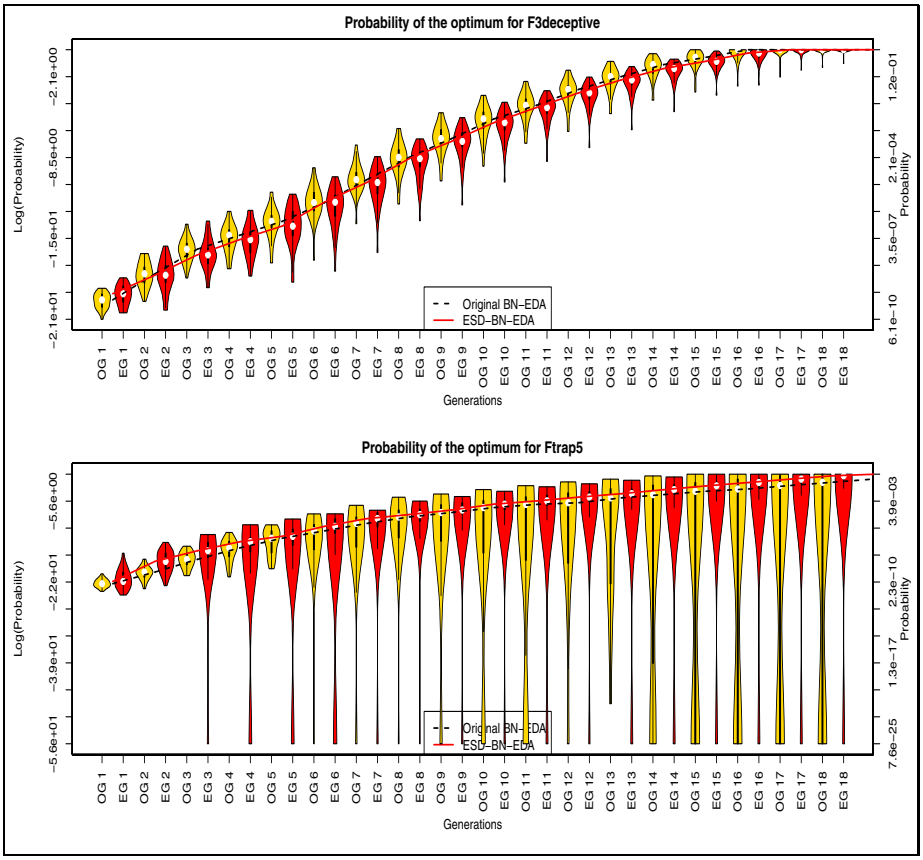

Fig. 9. Probability of sampling the optimum the truncation selection

to the last generations of the algorithms, when the learning of correct edges is not needed because the algorithm has converged.

Figure 4 also shows that the more complex the problem is ($f_{3deceptive}$ =3 correlated variables, $f_{trap5} = 5$ correlated variables), the greater is the evidence to say that the ESD-BN-EDA captures more correct edges than the original.

Figures 5 shows that for proportional selection there is a lot of strong evidence to say that the ESD-BN-EDA captures more correlations. And finally, 6 shows that for truncation selection we can not say which algorithm is better (a lot of $p-values >> 0$), hence as expected, for the truncation selection both algorithms perform quite similar.

4.2 Evidencing the Probability of Finding the Optimum

Recall that the goal of these experiments is not to show the effectiveness of the BN-EDA, which has been largely tested [17,8,20,19], in contrast we intend to show that the BN-EDA can be enhanced by using the ESD. An obvious

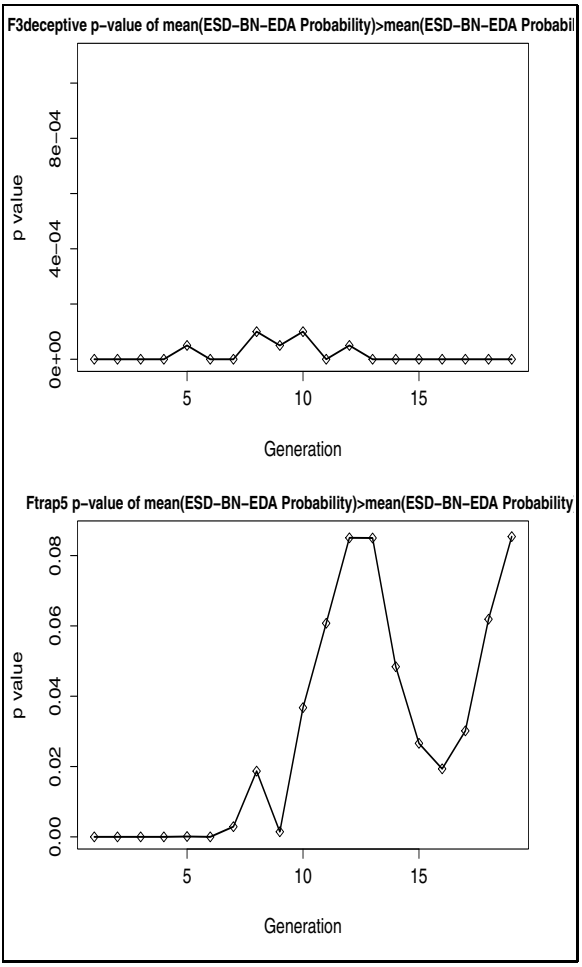

Fig. 10. $p - value$ for hypothesis test about the mean of the probability of sampling the optimum for the binary tournament

enhancement is to increase the probability of finding the optimum. Each generation is a different learning stage, because a cumulative bias is introduced in the model via the selection method, because of this, we test the probability of finding the optimum each generation in order to show a consistent increment of it when using the ESD. Using the same settings than in the experiment above, we compute for each generation the probability of finding the optimum. The findings are reported in Figures 7, 8 and 9. These Figures show the violin plots for the probability of the optimum, according to the computed structure and conditional probabilities of the Bayesian network, using the original BN-EDA and the ESD-BN-EDA, for 30 independent runs. In addition, the median of this probability is plotted by using a solid line for the ESD-BN-EDA and a dashed

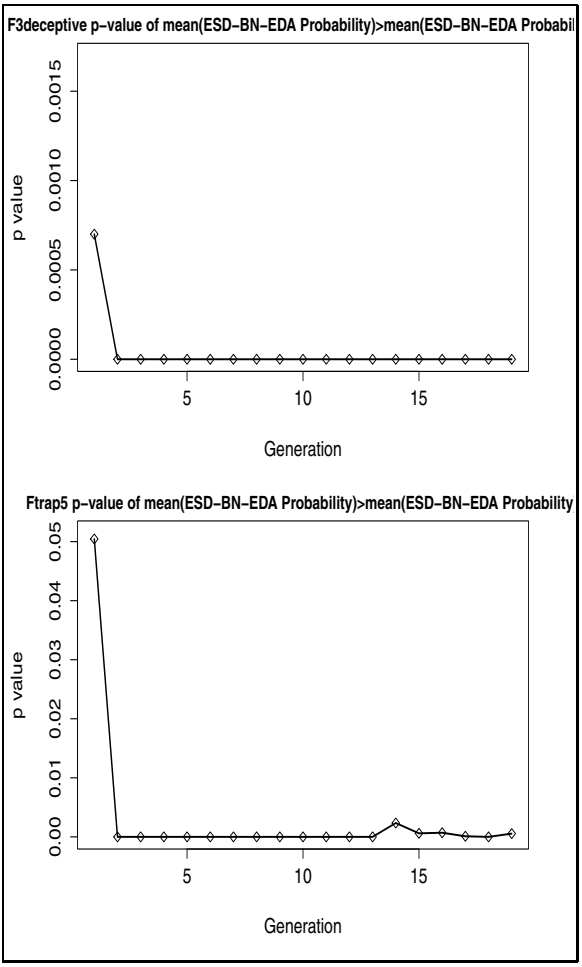

Fig. 11. $p - value$ for hypothesis test about the mean of the probability of sampling the optimum for the proportional selection

line for the original, the x coordinate is the same for both algorithms thus the lines reflect the actual difference between the medians. As can be seen there is evidence to say that the ESD-BN-EDA has a higher probability of finding the optimum during the whole generations. In order to show that the statistical evidence is strong enough we perform Boostrap hypothesis tests. The Figures 10, 11 and 12 are the p-value of testing that the probability of the optimum using the ESD is greater than the original counterpart.

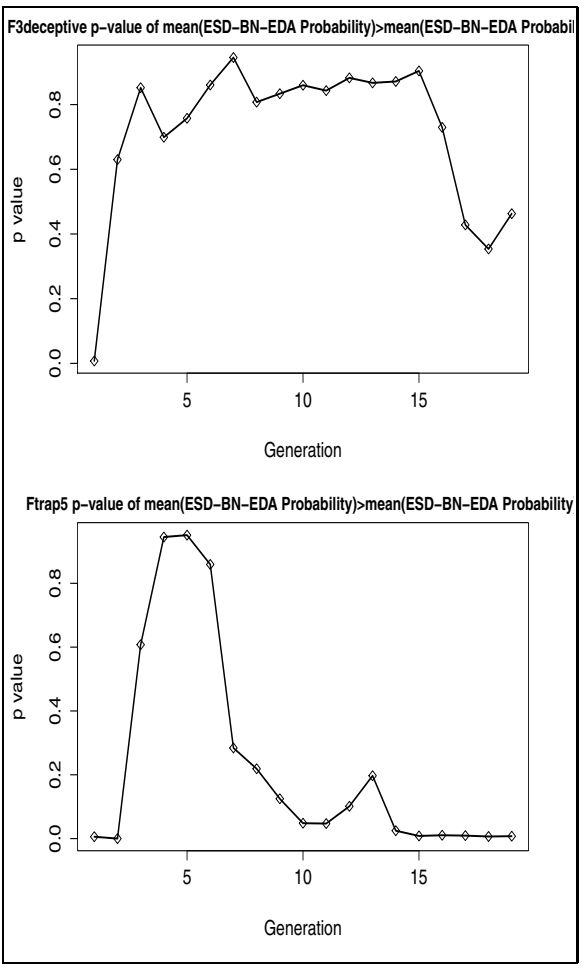

Fig. 12. $p - value$ for hypothesis test about the mean of the probability of sampling the optimum for the truncation selection

A Numerical Issue of This Test. Suppose that, according to the BN discovered, x_i depends on x_j at generation t. Assume the frequencies of the instances as follows: $freq(x_i = 0, x_j = 0) = 50$, $freq(x_i = 0, x_j = 1) = 50$, $freq(x_i = 1, x_j = 0) = 100$ and $freq(x_i = 1, x_j = 1) = 0$. Using the Bayes rule $p(x_i = 1|x_j = 1)$ is not defined, and the empirical joint probability of $p(x_i = 1, x_j = 1) = 0$. But notice that the empirical marginal probability, $p(x_i = 1) = 100/200 = 0.5$ and $p(x_j = 1) = 50/200 = 0.25$. Thus if the same frequencies (or quite similar) are maintained for the next generation, and x_i does not depend on x_j (given the new computed structure), then $p(x_i = 1, x_j = 1) = p(x_i)p(x_j) = (0.5)(0.25) = 0.125 > 0$. In practical terms this means that in a

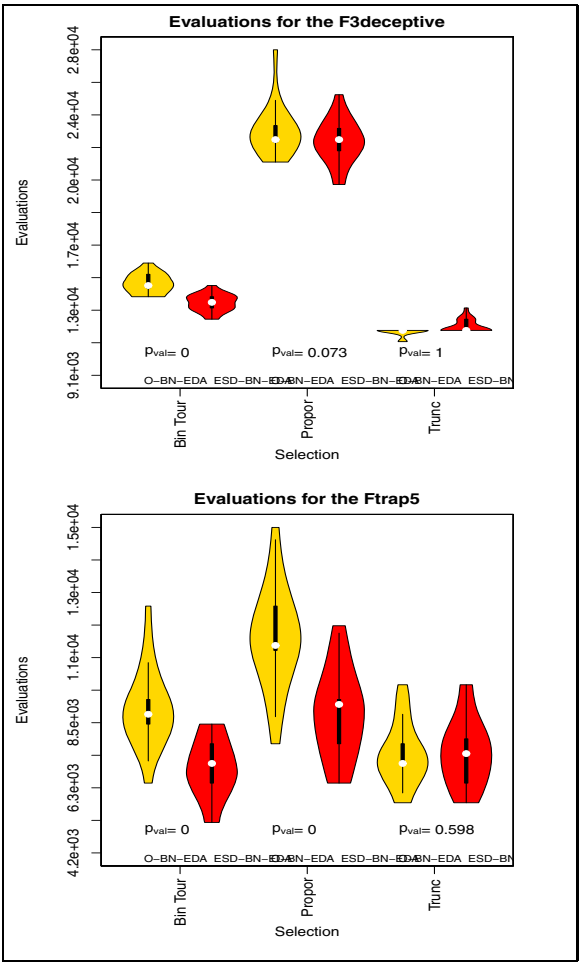

Fig. 13. $p - value$ and violin plots for comparison of the number of evaluations, for different selection methods. The hypothesis tested is mean(evaluations of BN-EDA)>mean(evaluations of ESD-BN-EDA).

generation the probability of sampling the optimum (under a given structure) could be 0, and some generations later the probability could be different from 0. Thus, for practical comparisons (because the violin plots of the probability are in log scale) we replace the probabilities equal to 0, with the worst value found in the same run, this only affects the plots (not the hypothesis test, neither the conclusions of the experiment) by avoiding to graphically report $-\infty$ values.

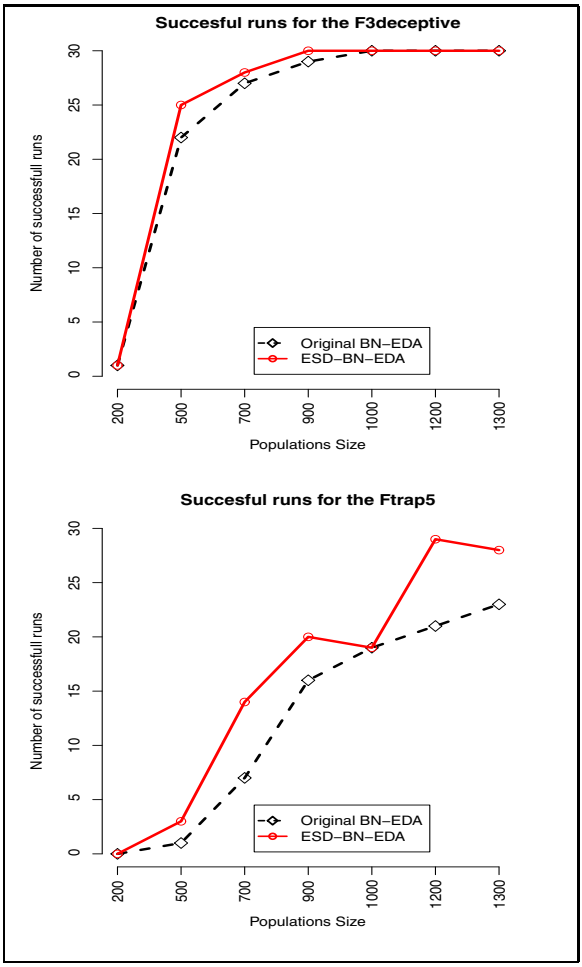

Fig. 14. Succesful runs from 30, for different population sizes using the binary tournament selection

Results of the Experiment Which Test the Probability of Finding the Optimum. As can be seen in Figures 7 to 12 there is sufficient statistical evidence to say that the ESD-BN-EDA has a greater probability to find the optimum than the original. The explanation is that the most fitted solutions actually are sharing variable values with the optimum and they have a greater weight to compute the Bayesian network structure and the conditional probabilities. The p-value close to 0 in the first generations is an remarkable indicator, because this stage of the algorithm is crucial for detecting the interest region where the optimum is. In this stage the EDA is performing a more intense exploration than in the last generations in which the algorithm basically is refining the optimum approximation and converging to a stable point. As expected the hypothesis

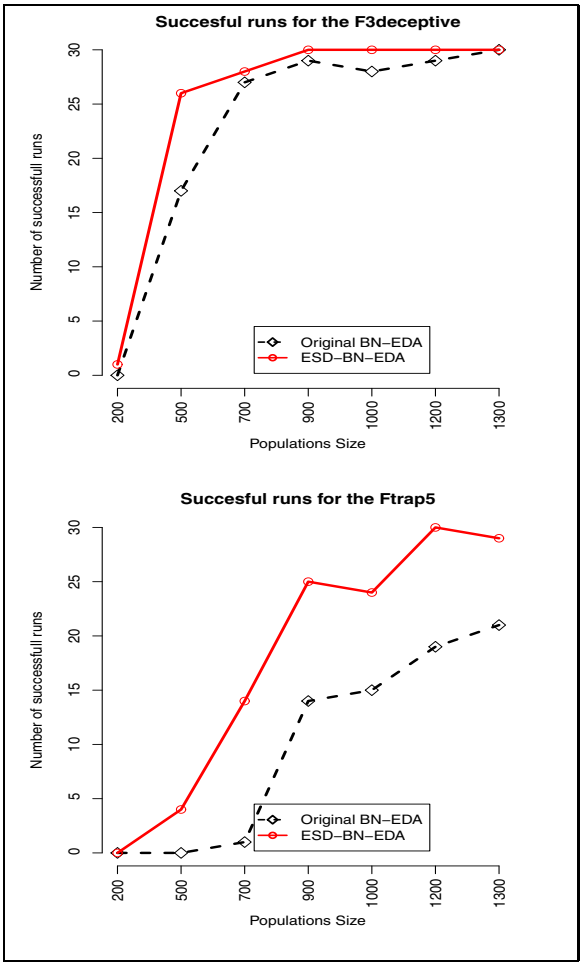

Fig. 15. Succesful runs from 30, for different population sizes using the proportional selection

test for the truncation method are not conclusive, because as mentioned for this selection the ESD-BN-EDA and the original are using the same information.

4.3 Evidencing the Reduction of the Number of Function Evaluations

In this experiment we run the BN-EDA and the ESD-BN-EDA, under the same parameters than the experiments above, but the algorithm was stopped when it finds the optimum. Then we present similar violin plots than in the experiments above, for the number of evaluations needed for each of the algorithms to reach the optimum. We use the same settings than the experiment above, except the

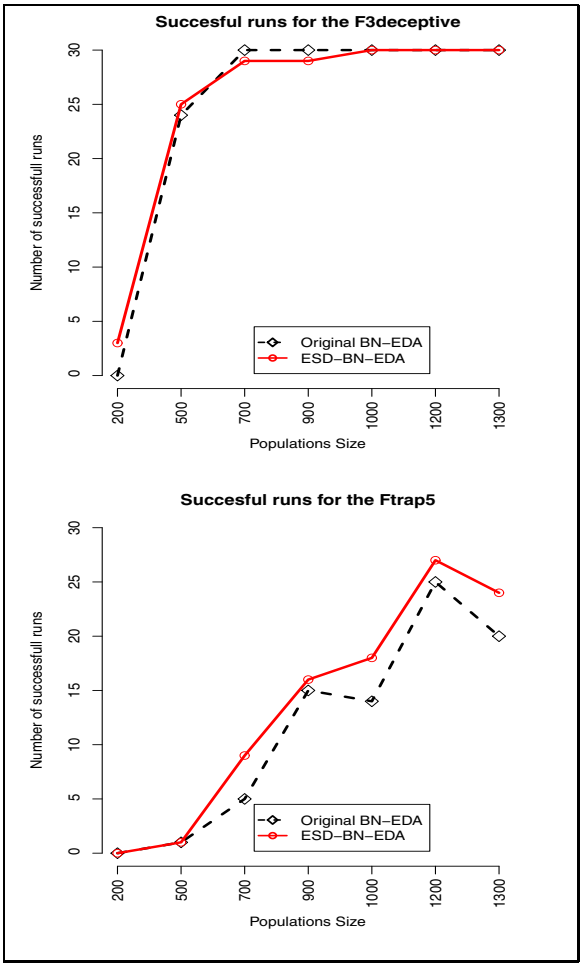

Fig. 16. Succesful runs from 30, for different population sizes using the truncation selection

population size which is fixed in 1300. 20 successful runs were used for the violin plots and the hypothesis tests. We tested if the mean of the number of evaluations of the original BN-EDA is greater than ESD-BN-EDA counterpart.

Results of the Experiment Which Test the Number of Evaluations. As can be seen the number of evaluations is less for the ESD-BN-EDA than the original one. And it is sufficient statistical evidence to support this conclusion. Additionally, the number of evaluations are not statistically different for the truncation method.

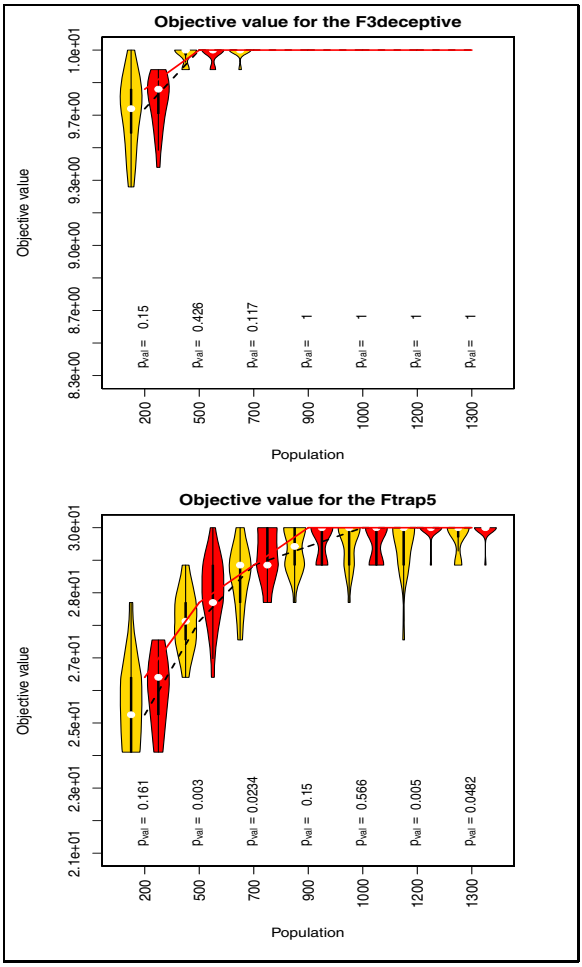

Fig. 17. Objective value for different population sizes for the binary tournament selection. The p-value correspond to the hypothesis test mean(f_{best} of ESD-BN-EDA) >mean(f_{best} of BN-EDA).

4.4 Evidencing the Reduction of the Population Size

In this experiment we use different population sizes and observe two different results:

1. The number of times the optimum is reached with each population.
2. The best objective value (in distribution via violin plots) found with each population. Also we perform hypothesis test to know if there is sufficient statistical evidence to say that the one algorithm delivers a better objective value.

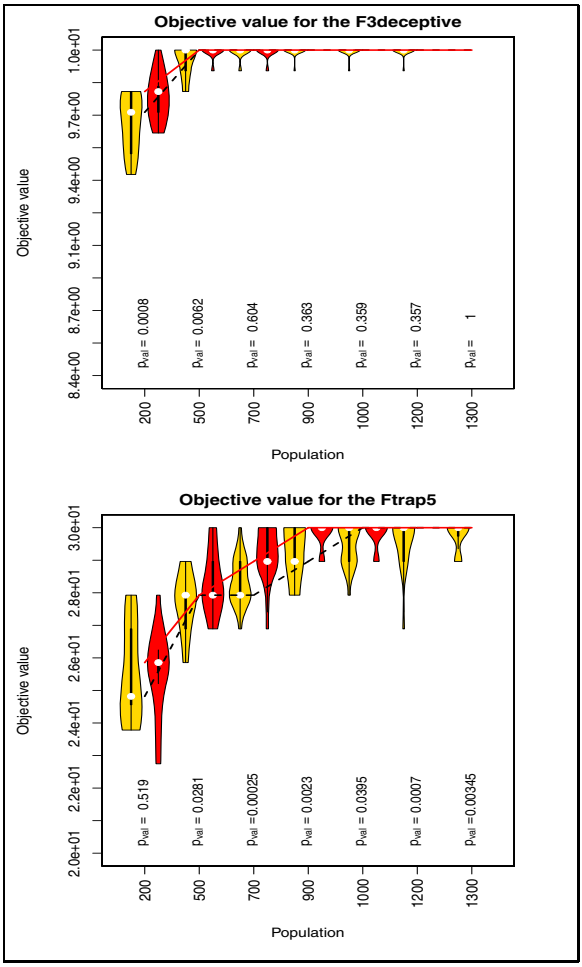

Fig. 18. Objective value for different population sizes for the proportional selection. The p-value correspond to the hypothesis test mean(f_{best} of ESD-BN-EDA) >mean(f_{best} of BN-EDA).

Results of the Experiment Which Test the Population Sizes. Figures 14, 15 and 16 show the number of times the optimum is reached with different population sizes, contrasting the original BN-EDA with ESD-BN-EDA. As can be see, the ESD-BN-EDA consistently outperform the BN-EDA. Figures 17, 18 and 19, show the objective function values of the elite individual for both approaches. Even if we do not always have sufficient statiscal evidence according to the p-value, it can be seen that many times the performance of the ESD-BN-EDA is the best, and in other cases it is a least as good as the original approach.

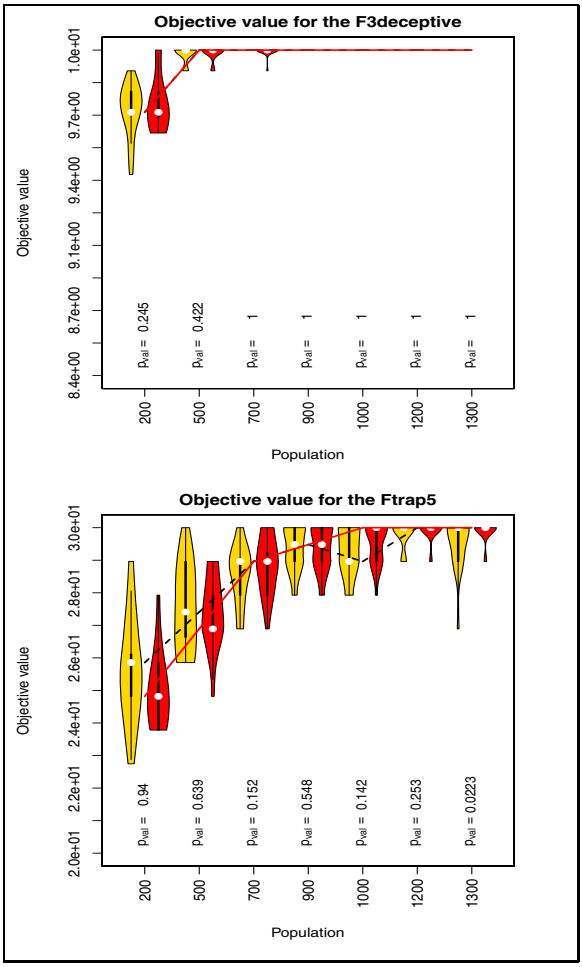

Fig. 19. Objective value for different population sizes for the truncation selection. The p-value correspond to the hypothesis test mean(f_{best} of ESD-BN-EDA) >mean(f_{best} of BN-EDA).

5 Conclusions

In this paper we deeply test the Empirical Selection Distribution (ESD) integrated in the selection-estimation step of the Bayesian Network based EDA(BN-EDA). According to our experiments the ESD significantly boosts the BN-EDA. The main conclusions are when using the Binary tournament and the proportional selection we collect sufficient statistical evidence to say that the ESD improves the BN-EDA performance. Several enhancements have been proposed for the BN-EDA [11,7,13], in this context the ESD enhances the BN-EDA, and at the same time, it is allowed to be combined with the other enhancements.

The ESD can be seen as a general selection method for biasing EDAs, in this article we show that even for complex models such as Bayesian Networks, integrating the ESD in an EDA can be relatively simple. Additionally, we test the ESD with three selection methods but notice that in general the ESD are weights for each individual. Then, other kind of bias such as Pareto raking for multi-objective approaches, or diversity measures can be used, in order to solve multiobjective problems, or to increase the impact of diverse solutions in the parameter estimation. In summary, the ESD enhances the BN-EDA but is not restricted to be used in it. Additionally, the ESD is not only a enhancement for an EDA but a way of inducing the bias in the search, hence researchers could propose different biasing schemes only by modifying the ESD computation, and maintaining unaltered the main EDA body. The results obtained by the boosting of the BN-EDA with the ESD, encourage the future work in using it with other multivariate discrete and continuous EDAs. Future work considers to integrate non-standard selection methods in the BN-EDA, for instance, methods which consider diversity measures. Additionally, we will continue exploring the effects of the ESD in other EDAs.

References

1. The Factorized Distribution Algorithm for additively decomposed functions, vol. 1 (1999)
2. Baluja, S.: Population-based incremental learning. Tech. Rep. CMU-CS-94-163, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA (June 1994)
3. Bonet, J.S.D., Isbell Jr., C.L., Viola, P.A.: MIMIC: Finding optima by estimating probability densities. In: Advances in Neural Information Processing Systems 9, NIPS, pp. 424–430. MIT Press (1996)
4. Efron, B.: The Jacknife, the Bootstrap and Other Resampling Plans. Society for Industrial and Applied Mathematics, 1400 Architect's Building, 117 South 17th Street, Philadelphia, Pensilvania (1982)
5. Harik, G., Goldberg, D.E.: Learning linkage. In: Proceedings of the 4th Workshop on Foundations of Genetic Algorithms, pp. 247–262 (1996)
6. Harik, G.R., Lobo, F.G., Goldberg, D.E.: The compact genetic algorithm. IEEE Trans. Evolutionary Computation 3(4), 287–297 (1999)
7. Hauschild, M.W., Pelikan, M., Sastry, K., Goldberg, D.E.: Using previous models to bias structural learning in the hierarchical boa. Evolutionary Computation 20(1), 135–160 (2012)
8. Hauschild, M., Pelikan, M., Sastry, K., Lima, C.: Analyzing probabilistic models in hierarchical boa. Trans. Evol. Comp. 13(6), 1199–1217 (2009)
9. Heckerman, D.: A tutorial on learning with Bayesian networks. Tech. Rep. MSR-TR-95-06, Microsoft Research, Advanced Technology Division, Microsoft Corporation (1995)
10. Heckerman, D., Geiger, D., Chickering, D.M.: Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. Machine Learning 20(3), 197–243 (1995)
11. Lima, C., Lobo, F., Pelikan, M., Goldberg, D.: Model accuracy in the bayesian optimization algorithm. Soft Computing - A Fusion of Foundations, Methodologies and Applications 15, 1351–1371 (2011)

12. Lima, C.F., Pelikan, M., Goldberg, D.E., Lobo, F.G., Sastry, K., Hauschild, M.: Influence of selection and replacement strategies on linkage learning in boa. In: IEEE Congress on Evolutionary Computation, pp. 1083–1090 (2007)
13. Luong, H.N., Nguyen, H.T.T., Ahn, C.W.: Entropy-based efficiency enhancement techniques for evolutionary algorithms. Information Sciences 188, 100–120 (2012)
14. Mühlenbein, H., Paaß, G.: From recombination of genes to the estimation of distributions I. Binary parameters. In: Voigt, H.M., Ebeling, W., Rechenberg, I., Schwefel, H.P. (eds.) PPSN 1996. LNCS, vol. 1141, pp. 178–187. Springer, Heidelberg (1996)
15. Mühlenbein, H.: Convergence theorems of estimation of distribution algorithms. In: Shakya, S., Santana, R. (eds.) Markov Networks in Evolutionary Computation. ALO, vol. 14, pp. 91–108. Springer, Heidelberg (2012)
16. Mühlenbein, H., Mahnig, T.: FDA -a scalable evolutionary algorithm for the optimization of additively decomposed functions. Evolutionary Computation 7(4), 353–376 (1999)
17. Pelikan, M., Goldberg, D.E., Cantú-Paz, E.: BOA: The Bayesian Optimization Algorithm. In: Banzhaf, W., Daida, J., Eiben, A.E., Garzon, M.H., Honavar, V., Jakiela, M., Smith, R.E. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference GECCO 1999, vol. I, pp. 525–532. Morgan Kaufmann Publishers, San Fransisco (1999)
18. Pelikan, M., Mühlenbein, H.: The Bivariate Marginal Distribution Algorithm. In: Advances in Soft Computing – Engineering Design and Manufacturing, pp. 521–535 (1999)
19. Pelikan, M., Sastry, K., Goldberg, D.E.: Scalability of the bayesian optimization algorithm. International Journal of Approximate Reasoning 31(3), 221–258 (2002)
20. Pelikan, M., Sastry, K., Goldberg, D.E.: iBOA: the incremental bayesian optimization algorithm. In: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, GECCO 2008, pp. 455–462. ACM, New York (2008)
21. Santana, R.: A Markov Network Based Factorized Distribution Algorithm for Optimization, pp. 337–348 (2003)
22. Shapiro, J.L.: Diversity loss in general estimation of distribution algorithms. In: Runarsson, T.P., Beyer, H.-G., Burke, E.K., Merelo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) PPSN 2006. LNCS, vol. 4193, pp. 92–101. Springer, Heidelberg (2006)
23. Valdez-Peña, S.I., Hernández-Aguirre, A., Botello-Rionda, S.: Approximating the search distribution to the selection distribution in EDAs. In: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, GECCO 2009, pp. 461–468. ACM, New York (2009)
24. Zhang, Q., Muhlenbein, H.: On the convergence of a class of estimation of distribution algorithms. Trans. Evol. Comp. 8(2), 127–136 (2004)