Statistical Properties of Differential Evolution and Related Random Search Algorithms

Daniela Zaharie

Faculty of Mathematics and Computer Science, West University of Timişoara
bv. Vasile Pârvan, no. 4, 300223 Timişoara, Romania, *dzaharie@info.uvt.ro*

Abstract. The aim of this paper is to analyze the impact on the expected population mean and variance of several variants of mutation and crossover operators used in differential evolution algorithms. As a consequence of this analysis a simple variance based mutation operator which does not use differences but has the same impact on the population variance as classical differential evolution operators is proposed. A preliminary analysis of the distribution probability of the population in the case of a differential evolution algorithm for binary encoding is also presented.

Keywords: differential evolution, population variance, mutation and crossover operators, premature convergence, binary encoding

1 Introduction

The analysis of the population dynamics induced by evolutionary operators is an important issue in understanding the behavior of evolutionary algorithms and in inferring rules about choosing adequate operators and control parameters. There are two main approaches in analyzing the dynamics of an evolutionary algorithm (Okabe (2005)): a cumulants based approach which tries to describe the dynamics by using cumulants (e.g. mean, variance etc.) and a model based approach which tries to build a probability model of the population based on the properties of the operators. Most results were obtained in the case of mutation operators based on normally distributed additive perturbations (Beyer (1998)). In the case of other evolutionary operators similar studies are significantly fewer. This is also the case of Differential Evolution (DE), a successful stochastic heuristic for global optimization for which the theoretical results on the impact of operators on the population properties are still limited. DE was introduced in (Storn and Price (1995)) and is based on a particular way of constructing so-called mutant vectors by using differences between randomly selected elements from the current population. Unlike stochastic mutation, typical to evolution strategies, the DE mutation uses only information extracted from the current population. For each mutant vector, a trial vector is constructed through a crossover operation. This trial vector competes with the corresponding element of the current population and the best one, with respect to the objective function, is transferred into the next generation. In the following we shall consider objective functions,

$f : D \subset \mathbb{R}^n \to \mathbb{R}$, to be minimized thus we are dealing with minimization problems of size n. The overall structure of DE (see Algorithm 1) is typical for evolutionary algorithms, the particularities of the algorithm being related with the mutation and crossover operators. By combining different DE mutation and crossover operators various schemes have been designed. In the DE literature these schemes are specified by using the convention $DE/a/b/c$ where a denotes the manner of constructing the mutant vector, b denotes the number of differences involved in the construction of the mutant vector and c denotes the crossover type.

```
Population initialization: $X(0) \leftarrow \{x_1(0),\dots,x_m(0)\}$
$g \leftarrow 0$
while the stopping condition is false do
  for $i = \overline{1,m}$
    $y_i \leftarrow$ generateMutant($X(g)$)
    $z_i \leftarrow$crossover($x_i(g)$,$y_i$)
    if $f(z_i) < f(x_i(g))$ then $x_i(g+1) \leftarrow z_i$ else $x_i(g+1) \leftarrow x_i(g)$
  endfor
  $g \leftarrow g+1$
endwhile
```

Fig. 1. The overall structure of a generational DE

Previous work on analyzing DE behavior by using a model-based approach is presented in Xue et al. (2005) and in Ali and Fatti (2006). Xue et al. analyze the impact of mutation on the population distribution starting from the assumption that the population current has a normal distribution. On the other hand, Ali and Fatti derive a rather sophisticated distribution probability which corresponds to the offspring obtained by mutation starting from a population uniformly distributed in the search space. The cumulants based approach is used in Zaharie (2002) where the influence of DE mutation and binomial crossover on the expected population variance is analyzed.

The main aim of this paper is to extend the results presented in Zaharie (2002) and in Zaharie (2007) for other crossover variants and to analyze a simple variance based mutation. The next section presents the mutation and crossover operators involved in the analysis while the main theoretical results are presented in Section 3. A variance based mutation having a behavior similar to DE/rand/1/* with respect to the impact on the population variance is presented in Section 4. Section 5 presents some preliminary results on a DE for binary encoding and Section 6 concludes the paper.

2 Differential evolution operators

2.1 Mutation operators

Mutation in differential evolution algorithms has the role of constructing mutant vectors by perturbing elements of the current population. The main particularity of DE mutation is the fact that the perturbation term is related with the difference between some randomly selected elements. Such a difference based mutation operator is more related to a recombination than to a classical mutation operator. Its main property is the fact that it acts as a self-referential mutation allowing a gradual exploration of the search space. The general form of the standard DE mutation is:

$$y_i = \lambda x_* + (1-\lambda)x_{I_i} + \sum_{l=1}^{L} F_l \cdot (x_{J_{il}} - x_{K_{il}}), \qquad i = \overline{1,m} \tag{1}$$

where x_* is the best element of the current population, $\lambda \in [0,1]$ is a coefficient which controls the influence of the best element, L is the number of differences, $F_l > 0$ is for each $l \in \{1,\dots,L\}$ a scaling factor. I_i, J_{il} and K_{il} are random values uniformly selected from $\{1,\dots,m\}$ and such that they are distinct. Most frequently used particular cases are when $L = 1$ and $\lambda \in \{0,1\}$. Thus for $\lambda = 0$ one obtains the DE/rand/1/* variant:

$$y_i = x_{I_i} + F \cdot (x_{J_i} - x_{K_i}), \qquad i = \overline{1,m} \tag{2}$$

and for $\lambda = 1$ one obtains the DE/best/1/* variant:

$$y_i = x_* + F \cdot (x_{J_i} - x_{K_i}), \qquad i = \overline{1,m}. \tag{3}$$

Other simple variants of these mutation operators are obtained by replacing the constant F with a random variable, ξ. Examples of such variants are when $\xi \sim N(0,F)$ (in Abbas (2001)), $\xi \sim N(F,\sigma)$ (in Ronkkonen (2003)) or even $\xi \sim U[F_{min}, F_{max}]$.

2.2 Crossover operators

In Evolutionary Algorithms the crossover operator usually combines features from different parents. In the case of DE algorithms, since the mutation operator is already based on a recombination of individuals, the role of crossover is somewhat different. It just allows the construction of an offspring by combining the current element and that generated by mutation. This can be ensured either by mixing the components (as in binomial and exponential DE crossover) or by an arithmetical recombination between the current and the mutant elements (as in the DE/current-to-rand variants). In the case of binomial crossover the components of the trial element z_i are obtained as:

$$z_i^j = \begin{cases} y_i^j & \text{if } U_j(0,1) < CR \text{ or } j = j_0 \\ x_i^j & \text{otherwise} \end{cases}, \qquad i = \overline{1,m}, j = \overline{1,n} \tag{4}$$

where $U_j(0,1)$ is a random value uniformly distributed in $[0,1]$, j_0 is a randomly selected value from $\{1,\dots,n\}$ and $CR \in [0,1]$ is a parameter which controls the number of components taken from the mutant vector, y_i, and is called crossover rate. The number of components taken form the mutant vector follows a binomial distribution of parameters n and $p_m = CR(1-1/n)+1/n$. The value p_m can be interpreted as mutation probability as long as it specifies the probability for a component to be taken from the mutant vector.

In the exponential crossover the trial vector is constructed by taking consecutive components from the mutant:

$$z_i^j = \begin{cases} y_i^j & \text{for } j \in \{j_0, \langle j_0+1\rangle_n, \ldots, \langle j_0+L-1\rangle_n\} \\ x_i^j & \text{otherwise} \end{cases} \tag{5}$$

In eq. (5) j_0 is a randomly selected index, $\langle j\rangle_n$ denotes the remainder of the division of j by n plus 1 and L is a random variable which follows a truncated geometric distribution (Zaharie (2007)). In this case the mutation probability satisfies:

$$p_m = \frac{1-CR^n}{n(1-CR)} \tag{6}$$

The arithmetical recombination consists in a convex combination of the current and mutant vector. Thus

$$z_i = (1-q)x_i + qy_i, \qquad i = \overline{1,m} \tag{7}$$

with $q \in [0,1]$ controlling the relative weight of the mutant vector. In some implementations (see for instance Mezura et.al (2006)) the arithmetical crossover is used just as a second step in generating the mutant vector while the trial vector is obtained by mixing the components of the vector given in Eq. (7) with the current element using binomial or exponential crossover.

3 Influence of mutation and crossover on the population mean and variance

As a result of the application of evolutionary operators the population changes its distribution. The parameters of the population distribution, especially the mean and variance, can give information about the region in the search space where the population is concentrated and about its diversity. A population can be interpreted as a set of random vectors, but since all components are evolved based on the same rule the analysis can be conducted componentwise. In the following we shall analyze the impact of several DE mutation and crossover operators on the population mean and variance. Let us denote by $\{X_1, \ldots, X_m\}$ the current population and by $\{Z_1, \ldots, Z_m\}$ the population obtained after mutation and crossover. Each element of this population is a random variable $Z_i = Y_i \cdot \mathbf{1}_{M_i} + X_i \cdot \mathbf{1}_{\overline{M_i}}$, with $\mathbf{1}_{M_i}$ denoting the indicator function corresponding to the event that Z_i equals the mutant element,

Y_i. Thus $\mathbf{1}_{M_i}$ is a random variable with the mean $E(\mathbf{1}_{M_i}) = p_m$. Similarly $E(\mathbf{1}_{\overline{M}_i}) = 1 - p_m$. The difference between binomial and exponential crossover is given only by different means of $\mathbf{1}_{M_i}$ and $\mathbf{1}_{\overline{M}_i}$.

As mutation operators we shall analyze the following variants:

$$Y_i = \lambda X_* + (1-\lambda) X_{I_i} + \sum_{l=1}^{L} \xi_l \cdot (X_{J_{il}} - X_{K_{il}}) \tag{8}$$

and

$$Y_i = (1-\eta) X_i + \eta X_{I_i} + \xi \cdot (X_{J_i} - X_{K_i}) \tag{9}$$

In eq. (8) X_* denotes the best element of the current population, $\lambda \in [0,1]$ and ξ_l denote random variables independent with respect to all other variables. The most known case is when $L = 1$ and ξ is constant and equal to the scaling factor, F. If $\lambda = 0$ one obtains the DE/rand/1/* variant and when $\lambda = 1$ one have the DE/best/1/* variant.

In eq. (9), η is usually a random variable on $[0,1]$. This variant is related both to current-to-rand variants and to those which use arithmetical recombination (in the case when $\eta = q$ and $\xi = q \cdot F$). In both cases I_i, J_i and K_i are uniformly distributed on $\{1, ...m\}$ and have distinct values.

In the following we shall estimate the expected mean and variance of the population obtained by applying mutation and crossover. The expected mean, $E(\overline{Z})$, of a population $\{Z_1, \dots, Z_m\}$ of identically distributed random variables equals $E(Z_i)$ for an arbitrary i. Thus $E(\overline{Z}) = E(Z_i) = p_m E(Y_i) + (1-p_m) E(X_i)$. Since for any random index I one have that $E(X_I) = E(\overline{X})$ it follows that in the case of eq. (8) one have that $E(\overline{Y_i}) = \lambda X_* + (1-\lambda) E(\overline{X})$, thus $E(\overline{Z}) = p_m \lambda X_* + (1 - p_m \lambda) E(\overline{X})$. Therefore when $\lambda = 0$ the expected population mean remains unchanged by mutation and crossover. When $\lambda > 0$ the population mean is biased toward the best element of the population. It is easy to check that the property of conserving the population mean is also true in the case of the mutation specified by eq. (9). The impact of selection depends on the objective function and is more difficult to be analyzed. However it is easy to see that after selection, the mean of the objective function values corresponding to the population elements decreases for all variants of mutation and crossover.

Let us turn now to the analysis of the expected population variance. Preserving the population diversity plays an important role in avoiding premature convergence and in stimulating the ability of differential evolution to follow dynamic optima. A natural measure of the diversity of a population of scalars, $X = \{X_1, \dots, X_m\}$, is the population variance $Var(X) = \sum_{i<j} (X_i - X_j)^2 / m^2$. In the case of populations of vectors the average of componentwise variances can be considered as a measure of diversity. In the following we shall analyze, in the one-dimensional case, the impact on the population variance of the mutation variants given by eqs. (8) and (9) com-

bined with binomial, exponential and arithmetical crossover. In all cases we estimate the expected population variance, $E(Var(Z))$.

Proposition 1. *The expected population variance after mutation and crossover is:*

$$E(Var(Z)) = \left(1 + 2p_m \sum_{l=1}^{L} E(\xi_l^2) - \tfrac{p_m(2-p_m)}{m} - \lambda p_m^2 \tfrac{m-1}{m}\right) E(Var(X))$$
$$+\lambda^2 p_m (1-p_m) \tfrac{m-1}{m} E((X_* - X_{I_i})^2) \qquad (10)$$

in the case of mutation operator given by (8) and

$$E(Var(Z)) = \Big(1 + 2p_m \left(E(\eta^2) - \tfrac{m-1}{m} E(\eta) + E(\xi^2)\right)$$
$$-\tfrac{p_m^2}{m}(2E(\eta) + E(\eta^2))\Big) E(Var(X)) \qquad (11)$$

in the case of mutation operator given by (9).

Proof. See Appendix.
In Proposition 1, p_m is given by $p_m = CR(1 - 1/n) + 1/n$ in the case of binomial crossover and by eq. (6) in the case of exponential crossover. Important particular cases of eq. (10) are when $\lambda = 0$ and $p_m = 1$. By denoting $F^2 = \sum_{l=1}^{L} E(\xi_l^2)$ we have in the first case:

$$E(Var(Z)) = \left(1 + 2p_m F^2 - \frac{p_m(2-p_m)}{m}\right) E(Var(X)) \qquad (12)$$

and in the second

$$E(Var(Z)) = \left((1-\lambda)\frac{m-1}{m} + 2F^2\right) E(Var(X)). \qquad (13)$$

When η is a constant q, $E(\xi^2) = F^2$ and $p_m = 1$ one obtains a simple current-to-rand version for which the eq. (11) becomes:

$$E(Var(Z)) = \left(1 + 2F^2 - 2q + \frac{2m-1}{m} q^2\right) E(Var(X)). \qquad (14)$$

If F^2 is replaced with $q^2 F^2$ then eq. (14) corresponds to the DE/rand/1 variant combined with arithmetical crossover. On the other hand, when η is uniformly distributed on $[0,1]$ and $p_m \in [0,1]$ then

$$E(Var(Z)) = \left(1 + 2p_m F^2 - \frac{1}{3m}(4p_m^2 + (m-3)p_m)\right) E(Var(X)). \qquad (15)$$

In almost all cases (except for the case when $\lambda > 0$ and $p_m < 1$) there is a simple linear relationship between the expected variance of the population

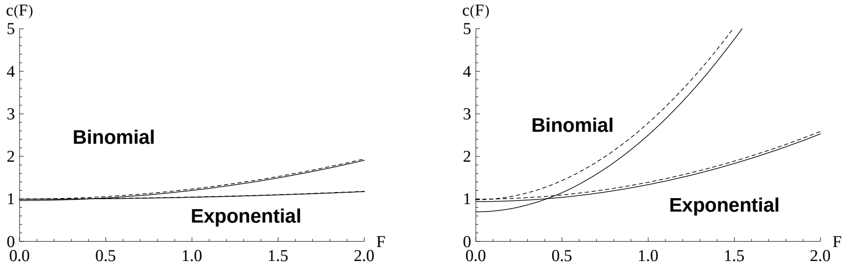

Fig. 2. Dependence between the variance factor, c, and F for DE/rand/1/* (dashed line) and DE/current-to-rand/1/* with $\eta \in U(0,1)$(continuous line). Parameters: $m = n = 50$, $CR = 0.1$ (left) and $CR = 0.9$ (right).

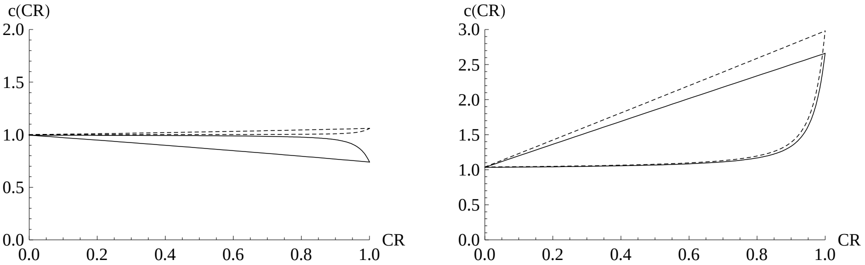

Fig. 3. Dependence between the variance factor,c, and CR for DE/rand/1/* (dashed line) and DE/current-to-rand/1/* with $\eta \in U(0,1)$ (continuous line). Binomial crossover: linear dependence, exponential crossover: nonlinear dependence. Parameters: $m = n = 50$, $F = 0.2$ (left) and $F = 1$ (right).

obtained by mutation and crossover and the variance of the current population: $E(Var(Z)) = c(CR, F, q, m, n)E(Var(X))$. The coefficient of this dependence involves all parameters which influence the algorithm behavior. The advantage of such a property is the fact that one can control the impact which mutation and crossover have on the population variance by changing the values of the parameters involved in c. Figures 2,3,4 and 5 illustrate the dependence of the factor $c(CR, F, q, m, n)$ on the values of parameters and on the algorithm type. The main remarks are: *(i)* c usually increases with CR and F but in a different way in the case of binomial and exponential crossover; *(ii)* the DE/current-to-rand variant is characterized by values of c slightly smaller than DE/rand; moreover, for small values of F (e.g. $F = 0.2$) c decreases when CR increases; *(iii)* the ratio m/n does not significantly influence the factor c, meaning that using larger populations does not stimulate the population diversity; *(iv)* both in the case of DE/best and DE/current-to-rand the variance is significantly increasing with the value of F but it decreases with λ and has a non-monotonous behavior with respect to q.

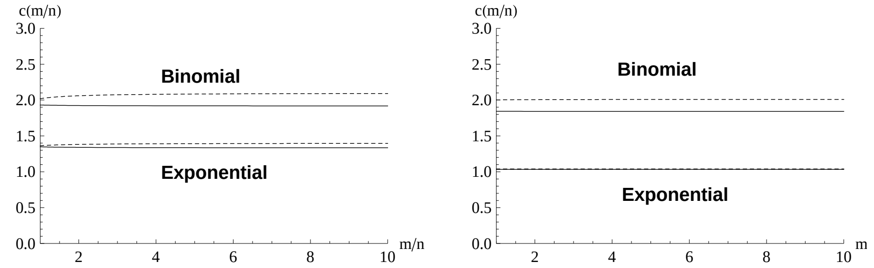

Fig. 4. Dependence of the variance factor, c, on the population size factor, s ($m = sn$) for DE/rand/1/* (dashed line) and DE/current-to-rand/1/* with $\eta \in U(0,1)$(continuous line). Parameters: $CR = 0.5$, $F = 1$, $n = 10$ (left) and $n = 100$ (right).

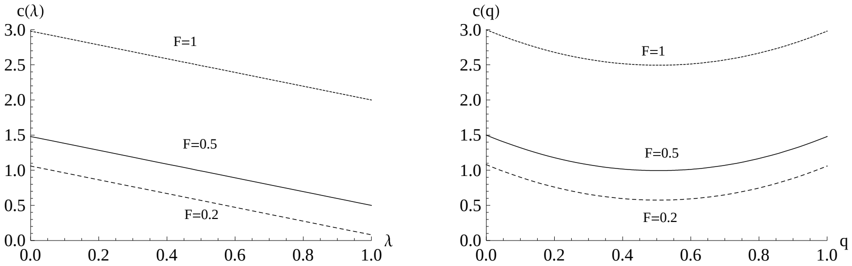

Fig. 5. Dependence of the variance factor, c, on the parameter λ in the case of DE/best-to-rand (left) and on parameter q in the case of DE/current-to-rand when $\eta = q$ (right). Parameters: $CR = 1$, $m = n = 50$.

4 A simple variance based variant

Classical mutation operators based on an additive perturbation also lead to a linear dependence but with a non zero free term, i.e. $E(Var(Z)) = aE(Var(X)) + b$. Let us consider the case when $Z_i = (X_{I_i} + \xi_i)1_{M_i} + X_i 1_{\overline{M_i}}$ with $E(M_i) = p_m$ and ξ_i independent random variables having $E(\xi_i) = 0$ and $E(\xi_i^2) = F^2$. In this case one obtains that $E(Var(Z)) = (1 + p_m^2/m - 2p_m/m)E(Var(X)) + 2p_m(m-1)/mF^2$. It is easy to see that if $E(\xi_i^2) = F^2 E(Var(X))m/(m-1)$ one obtains the same dependence between the expected variances as in the case of mutation given by eq. (8) with $\lambda = 0$ and $L = 1$. Thus using the mutation rule

$$y_i^j = x_{I_i^j} + F\sqrt{\frac{m}{m-1}Var(x^j)}N(0,1), \qquad j = \overline{1,n} \tag{16}$$

where $Var(x^j)$ is the variance of the current population corresponding to the jth component, one obtains a simple mutation rule which combined with a crossover strategy leads to the same behavior with respect to the evolution of the population variance as DE/rand/1/*. Empirical studies conducted for classical test functions show that the mutation given by eq. (16) behaves better than a simple evolution strategy based on a normal perturbation of distribution $N(0, F)$. Table 1 presents results obtained by 30 independent runs of the algorithms on Griewank test function for $n = 100$. The maximal number of function evaluations (nfe) is set to 500000 and a run is considered successful if the best value of the population (f^*) is less than $\epsilon = 10^{-8}$. These results illustrate the fact that for some pairs of values (CR, F) the variance based mutation leads to a behavior similar to that of DE. Thus by just ensuring that the population variance have the same dynamics one can partially reproduce the behavior of DE. On the other hand, the empirical results show that the difference-based perturbation cannot just be replaced with a variance-based perturbation since for other values of parameters or for other test functions the difference-based mutation leads to better results than the variance-based one.

5 Binary differential evolution

Encouraged by the success of DE in continuous optimization several authors recently proposed variants of DE for binary encoding (Gong (2006)). A simple approach is just to use the classical operators in order to evolve trial vectors in $[0,1]^n$ and transform their components in binary values, using a threshold function, only when the objective function is to be evaluated. The variant which we analyze uses a binary encoding and is based on the following mutation rule, inspired from (Gong (2006)):

$$Y_i^j = \begin{cases} x_{I_i}^j & \text{if } x_{J_i}^j = x_{K_i}^j \text{ or } U \geq F \\ 1 - x_{I_i}^j & \text{otherwise} \end{cases} \tag{17}$$

where $F \in [0,1]$ and U is a random value uniformly generated in $[0,1]$. The components of the trial vector, Z_i, are obtained by applying one of the DE crossover operators. Disregarding the type of crossover we shall denote by p_m the probability that a component in the trial vector is taken from the mutant vector. In the following we shall analyze the influence the mutation and crossover have on the distribution of a population of scalar elements.

Let (p_0, p_1) be the probability distribution of the current population (p_0 is the probability that a randomly selected element has the value 0). Thus for two randomly selected elements x_{K_i} and x_{J_i} we have that

$$Prob(x_{K_i} = x_{J_i}) = p_0^2 + p_1^2 \text{ and } Prob(x_{K_i} \neq x_{J_i}) = 2p_0p_1 \tag{18}$$

It follows that $Prob(Y_i = 0) = p_0(p_0^2 + p_1^2) + 2Fp_0p_1^2 + 2(1-F)p_0p_1^2$ and $Prob(Y_i = 1) = p_1(p_0^2 + p_1^2) + 2Fp_0^2p_1 + 2(1-F)p_0^2p_1$. Consequently, the probabilities corresponding to the trial element Z_i are $Prob(Z_i = 0) = p_0(1 + 2p_mFp_1(p_1 - p_0))$ and $Prob(Z_i = 1) = p_1(1 + 2p_mFp_0(p_0 - p_1))$. On the other hand, in the case of a simple binary mutation ($Z_i = 1 - X_i$ with probability

Table 1. Comparative results of DE/rand/1/bin, variance based mutation (var/bin) and normal mutation (norm/bin) combined with binomial crossover. Test function: Griewank. Parameters: $m = n = 100$.

CR	F	DE/rand/1/bin		var/bin		norm/bin	
		$\langle f^* \rangle$	Success	$\langle f^* \rangle$	Success	$\langle f^* \rangle$	Success
		$stdev(f^*)$	$\langle nfe \rangle$	$stdev(f^*)$	$\langle nfe \rangle$	$stdev(f^*)$	$\langle nfe \rangle$
0.1	0.5	$9 \cdot 10^{-9}$	30/30	$9 \cdot 10^{-9}$	30/30	0.3304	0/30
		$\pm 10^{-10}$	(380416)	$\pm 10^{-10}$	(190290)	±0.3134	(500000)
0.5	0.5	10^{-4}	0/30	$9 \cdot 10^{-9}$	30/30	0.2890	0/30
		$\pm 10^{-5}$	(500000)	$\pm 10^{-10}$	(204703)	±0.087	(500000)
0.9	0.5	0.0078	18/30	$1.27 \cdot 10^{-8}$	27/30	0.5523	0/30
		±0.0125	(306933)	$\pm 10^{-8}$	(470792)	±0.039	(500000)
0.1	0.2	$9 \cdot 10^{-9}$	30/30	0.0158	24/30	0.6352	0/30
		$\pm 2 \cdot 10^{-10}$	(137090)	0.0318	(131887)	±0.365	(500000)
0.5	0.2	0.0959	18/30	1.3469	0/30	0.4322	0/30
		±0.1657	(87666)	1.5373	(500000)	±0.3319	(500000)

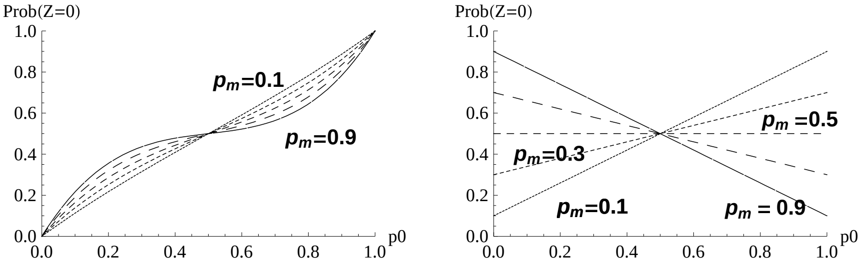

Fig. 6. Dependence of $Prob(Z_i = 0)$ on p_0 and p_m in the case of binary DE mutation (left) and classical binary mutation (right)

p_m) the corresponding probabilities are $Prob(Z_i = 0) = p_0 + p_m(p_1 - p_0)$ and $Prob(Z_i = 1) = p_1 + p_m(p_0 - p_1)$. The different impact on the population distribution of the DE binary mutation and of the classical binary mutation is illustrated in Figure 6. Unlike the classical binary mutation, the DE binary mutation leads to small changes in the population distribution for all values of p_m. On the other hand one have to remark that $Prob(Z_i = 0) - Prob(Z_i = 1) = (1 - 2p_m)(p_0 - p_1)$ in the case of classical binary mutation and $Prob(Z_i = 0) - Prob(Z_i = 1) = (1 - 2p_m F p_0 p_1)(p_0 - p_1)$ in the case of DE mutation. Both variants tend to decrease the difference between p_0 and p_1 but the decrease is smaller in the case of DE variant.

6 Conclusions

Almost for all DE variation operators the expected population variance after mutation and crossover is related to the current population variance by a simple linear dependence based on a coefficient $c(CR, F, q, m, n)$ which involves all parameters characterizing the algorithm. This allows us to control the evolution of the population diversity by just changing the algorithm's parameters. Significant differences have been identified between the behavior of binomial, exponential and arithmetical crossover. A simple mutation rule which does not involve differences but just the estimation of the current variance was proposed. It has the same behavior as DE/rand/1/* with respect to the population variance evolution. Numerical experiments show that for some sets of parameters the variance based mutation combined with binomial crossover behaves better than DE/rand/1/bin but for other ones worse. This suggests on one hand that the evolution of the population variance has a significant influence on the behavior of the algorithm and on the other hand that the difference based mutation induces a dynamics which cannot be entirely mimicked by using the population variance estimation instead. The analysis of the influence of the DE binary mutation on the population distribution shows that it leads to a dynamics different than that induced by classical binary mutation. However further analysis is needed to assess its effectiveness for real problems.

Appendix. *Proof of Proposition 1.* Since $E(Var(X)) = \dfrac{m-1}{2m} E((X_i - X_j)^2)$ for any pair of distinct indices (i, j) it follows that it is enough to find the relationship between $E((Z_i - Z_j)^2)$ and $E((X_i - X_j)^2)$ for an arbitrary pair (i, j) of distinct values. Based on the fact that $Z_i = Y_i \mathbf{1}_{M_i} + X_i \mathbf{1}_{\overline{M_i}}$ it follows:

$$\begin{aligned} E((Z_i - Z_j)^2) &= p_m^2 E((Y_i - Y_j)^2) + 2p_m(1 - p_m) E((Y_i - X_j)^2) \\ &+ (1 - p_m)^2 E((X_i - X_j)^2) \end{aligned} \tag{19}$$

If I and J are two random indices from $\{1, ..., m\}$ then $E((X_I - X_J)^2) = 2E((Var(X))$ if I and J can be identical and $E((X_I - X_J)^2) = \frac{2m}{m-1} E((Var(X))$ if I and J take distinct values. Using these relations one can compute $E((Y_i - Y_j)^2)$ and $E((Y_i - X_j)^2)$ when Y_i is given by eqs. (8) and (9).

By taking into account that I_i and I_j can be identical but K_{il} and J_{il}, K_{jl} and J_{jl} are respectively distinct one obtains, in the case of eq.(8) that

$$E((Y_i - Y_j)^2) = \frac{2m}{m-1}((1 - \lambda)^2 + 2\sum_{l=1}^{L} E(\xi_l^2)) E(Var(X))$$

and

$$E((Y_i - X_j)^2) = 2(1 + \frac{m}{m-1} \sum_{l=1}^{L} \xi_l^2) E(Var(x)) + \lambda^2 E((X_* - X_{I_i})^2)$$

By replacing these into eq. (19) one obtains eq. (10). In the case of eq.(9) one have that:

$$E((Y_i - Y_j)^2) = \frac{2m}{m-1} \left(E((1 - \eta)^2) + 2E(\xi^2) + \frac{m-1}{m} E(\eta^2) \right) E(Var(X))$$

and

$$E((Y_i - X_j)^2) = \frac{2m}{m-1}\left(1 + E(\eta^2) + E(\xi^2) - \frac{m-1}{m}E(\eta)\right) E(Var(X))$$

By replacing these into eq. (19) one obtains eq. (11).

Acknowledgments. Thanks are due to Olivier François for valuable suggestions on the analysis of the variance-based variant. This work is supported by Romanian project 11-028/14.09.2007.

References

ABBAS, H.A, SARKER, R. and NEWTON, C. (2001): A Pareto Differential Evolution Approach to the Vector Optimisation Problems. In: *Proceedings of CEC 2001, IEEE Computer Press, 971-978.*

ALI, M.M. and FATTI, L.P. (2006): A Differential Free Point Generation Scheme in the Differential Evolution Algorithm. *Journal of Global Optimization 35, 551-572.*

BEYER, H.G. (1998): On the Dynamics of EAs without Selection. In: W. Banzhaf and C.R. Reeves, *Proceedings of the Fifth Workshop on Foundations of Genetic Algorithms - FOGA 1998, 5-26*

GONG, T. and TUSON, A. (2007): Differential Evolution for Binary Encoding.In: Soft Computing in Industrial Applications, (Springer-Verlag,2007) vol. 39, 251-262.

MEZURA-MONTES, E., VELASQUEZ-REYES, J. and COELLO COELLO, C.A.(2006): A Comparative Study of Differential Evolution Variants for Global Optimization. In: Maarten Keijzer et al. (eds.) *Genetic and Evolutionary Computation Conference (GECCO'2006), Vol. 1, (ACM Press,2006) 485-492.*

OKABE,T., JIN, Y. and SENDHOFF,B. (2005): Theoretical Comparisons of Search Dynamics of Genetic Algorithms and Evolution Strategies. In: *Proceedings of CEC 2005, 382–389.*

RONKKONEN, J., LAMPINEN, J. (2003): On using normally distributed mutation steplength for the differential evolution algorithm. In: R. Matoušek and P. Ošmera (eds.): *Proc. of Mendel 2003, 9th International Conference on Soft Computing*, 11–18.

STORN, R. and PRICE, K. (1995): Differential Evolution a simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Science Institute, Berkeley, TR- 95-012.

XUE, F., SANDERSON, A.C. and GRAVES, R.J. (2005): Multi-objective differential evolution - algorithm, convergence analysis and applications. In: *Proceedings of CEC 2005, IEEE Computer Press, 743-750.*

ZAHARIE, D. (2002): Critical values for the control parameters of differential evolution algorithms. In: *R. Matoušek and P. Ošmera, eds., Proceedings of 8th International Conference on Soft Computing, Mendel 2002, 62-67.*

ZAHARIE D. (2007): A Comparative Analysis of Crossover Variants in Differential Evolution. In: *Proceedings of the IMCSIT, 2nd International Symposium Advances in Artificial Intelligence and Applications, 2007, 171-181.*