Effect of Multivariate Cauchy Mutation in Evolutionary Programming

Chang-Yong LEE\(^{(a)}\), Member and Yong-Jin PARK\(^{(b)}\), Nonmember

SUMMARY  In this paper, we apply a mutation operation based on a multivariate Cauchy distribution to fast evolutionary programming and analyze its effect in terms of various function optimizations. The conventional fast evolutionary programming in-cooperates the univariate Cauchy mutation in order to overcome the slow convergence rate of the canonical Gaussian mutation. For a mutation of \(n\) variables, while the conventional method utilizes \(n\) independent random variables from a univariate Cauchy distribution, the proposed method adopts \(n\) mutually dependent random variables that satisfy a multivariate Cauchy distribution. The multivariate Cauchy distribution naturally has higher probabilities of generating random variables in inter-variable regions than the univariate Cauchy distribution due to the mutual dependence among variables. This implies that the multivariate Cauchy random variable enhances the search capability especially for a large number of correlated variables, and, as a result, is more appropriate for optimization schemes characterized by interdependence among variables.

In this sense, the proposed mutation possesses the advantage of both the univariate Cauchy and Gaussian mutations. The proposed mutation is tested against various types of real-valued function optimizations. We empirically find that the proposed mutation outperformed the conventional Cauchy and Gaussian mutations in the optimization of functions having correlations among variables, whereas the conventional mutations showed better performance in functions of uncorrelated variables.

key words: fast evolutionary programming, mutation operation, multivariate Cauchy distribution, function optimization, inseparable functions

1. Introduction

In meta-heuristics for a continuous optimization problem, the variation of a state (or variable) is usually carried out by generating random variables from a predetermined probability distribution. A widely used distribution is a Gaussian probability distribution. In particular, the canonical evolutionary strategies (ESs)\[1\], [2] and evolutionary programming (EP)\[3\], [4], in which the mutation operation is the main operation, generate an offspring via mutations based on the Gaussian probability distribution. Recently, a mixed strategy\[5\] and an ensemble approach\[6\] were proposed to cope with the limitation of a single mutation operation. In addition, the Gaussian mutation has been refined with the covariance matrix adaptation, called CMA-ES\[7\], in which the mutation based on the Gaussian distribution is further correlated in terms of the adapted covariance matrix of the distribution throughout the evolution. The canonical EP and its variants have been successfully applied to engineering problems\[8\], [9].

It is known that the canonical EP has rather slow convergence rates in function optimizations\[10\], [11]. This is mainly due to the fact that the Gaussian mutation adopted in EP is basically a local sampling because of a finite variance of the Gaussian distribution, thus mutated offspring may not be very different from their parents. This slow convergence rate can also be true for any variation based on a bounded (or finite) variance distribution according to the central limit theorem\[12\]. The slow convergence rate was overcome, at least partially, if not entirely, by EP with mutations based on a univariate Cauchy or Lévy probability distribution, called the fast evolutionary programming (FEP)\[10\], [11]. This fast convergence is closely related to the characteristic of the Cauchy distribution. Due to infinite moments of the distribution, the distribution can occasionally generate a long jump among essentially local samplings over the search space. The proper trade-off between local and global searches allows a fast convergence to the solution. It was shown that these mutations had advantages over the conventional Gaussian mutation in real-valued function optimizations\[10\], [11].

In both EP and FEP, \(n\) variables are mutated by independently generated \(n\) random variables from either a Gaussian or a univariate Cauchy distribution. Independently generated random variables, however, are not necessarily suitable for a real-valued function optimization in which variables are correlated (or interdependent). Whether independently mutated variables properly take into account the correlations among variables depends on the underlying probability distribution. In the Gaussian mutation, for example, independently generated \(n\) random variables from a Gaussian distribution form a Gaussian multivariate random variable (or random vector) of \(n\) components. The same argument, however, cannot be applied to the case of the Cauchy mutation. That is, \(n\) random variables generated independently from a univariate Cauchy distribution do not form a multivariate Cauchy random variable. While independently generated univariate Cauchy random variables are suitable when \(n\) variables independently take part in the optimization process, they may search the variable space ineffectively when there exist correlations among variables. Therefore, in order to properly take into account the effect of correlation among variables while maintaining the merit of the Cauchy mutation over the Gaussian mutation, it is neces-
sary to have a scheme for generating a multivariate Cauchy random variable.

A multivariate Cauchy distribution was utilized to study the theoretical characteristics of the mutation, such as the convergence rate and the rate of progress [13], [14]. In Ref. [13], the local convergence behavior was studied by investigating the ability to locate a local minimum using the spherically symmetric property of a multivariate Cauchy distribution and the non-sphericity of a univariate Cauchy distribution. In Ref. [14], the rate of progress of an offspring was investigated in terms of the robustness and the probability of an offspring progressing by using the multivariate (2-dimensional) Cauchy distribution. These studies, however, did not introduce a method of generating a random vector from a multivariate Cauchy distribution.

In this paper, we propose a mutation operation based on a multivariate Cauchy distribution by introducing an algorithm for generating a random vector of a standard multivariate Cauchy distribution. We demonstrate the effect of the proposed mutation through empirical analysis. We also discuss under what conditions the proposed mutation has an advantage and/or disadvantage compared to the conventional ones. To this end, we test the proposed mutation against conventional mutations by the optimization of typical benchmarking functions.

It turns out that, the proposed multivariate Cauchy mutation (MCM) properly handles correlations among variables, thus it searches the variable space more effectively than the univariate Cauchy mutation (UCM). In addition, MCM can resolve, at least partially, the shortcoming of the Gaussian mutation (GM) in that GM is ineffective to explore a wider area of the search space. Thus, we expect that MCM outperforms the conventional mutations for functions having correlations among variables. In this sense, MCM has the advantages over both GM and UCM, while it maintains the advantage of UCM over GM.

This paper is organized as follows. In Sect. 2, we introduce a multivariate Cauchy probability distribution and an algorithm for generating multivariate Cauchy random variables. Section 3 describes FEP for multi-dimensional function optimization problems using all mentioned mutations. This is followed by the experimental results and discussion of the function optimization. The last section contains the conclusion and directions for future studies.

2. Multivariate Cauchy Distribution and Its Random Variable

A random variation \( \xi \) of \( n \) components is said to have the multivariate Cauchy distribution if every linear combination of its components has a univariate Cauchy distribution. The probability density function of the multivariate Cauchy distribution [15] is given as

\[
p (\vec{x}, \gamma) = \frac{\Gamma ((n+1)/2)}{\pi^{(n+1)/2}} \left( \frac{\gamma}{\vec{x}^2 + \gamma^2(n+1)/2} \right),
\]

where \( \gamma > 0 \) is the scale parameters and can be set to \( \gamma = 1 \) without a loss of generality. It should be noted that \( n \) components of \( \vec{x} \) are not statistically independent for any scale parameter \( \gamma \) [16]. This implies that independently generated \( n \) univariate Cauchy random variables cannot form an \( n \)-dimensional multivariate Cauchy random variable. Therefore, the crucial point in the generation of multivariate Cauchy random variables lies in the handling of the dependence among \( n \) components.

The generating a multivariate random variable is much more involved than that of a univariate one. The method adopted in this paper is the conditional distribution approach. More specifically, it is known that, given \( Z_1 = z_1, \ldots, Z_{m-1} = z_{m-1} \) having a univariate Cauchy distribution, the conditional distribution of \( Y_m (m = 2, 3, \ldots, n) \),

\[
Y_m = \left[ \sqrt{m} \left( 1 + \sum_{i=1}^{m-1} Z_i^2 \right)^{-1/2} \right] Z_m,
\]

is a Student’s \( t \)-distribution with \( m \) degrees of freedom [17]. An advantage of this method is that the problem of generating an \( n \)-dimensional multivariate Cauchy random variable is reduced to the generation of a series of \( n \) univariate Cauchy random variables and their appropriate transformation.

The algorithm for generating a standard (\( \gamma = 1 \)) multivariate Cauchy random variable of \( n \) components \((x_1, x_2, \ldots, x_n)\) can be described as follows.

**Step 1:** Generate \( n - 1 \) random variables of the univariate Cauchy distribution, \( z_1, z_2, \ldots, z_{n-1} \).

**Step 2:** Set \( x_1 = z_1 \).

**Step 3:** Iterate the following for \( x_m, m = 2, 3, \ldots, n \).

\[
x_m = m^{-1/2} \left( 1 + \sum_{i=1}^{m-1} z_i^2 \right)^{1/2} t(m),
\]

where \( t(m) \) is the random variable of a Student \( t \)-distribution with \( m \) degrees of freedom.

Note that the dependence among components of a multivariate Cauchy random variable is manifested by Eq. (3). The random variable of a Student \( t \)-distribution can be readily sampled by the following transformation:

\[
t(m) = \frac{Z}{\sqrt{Y/m}},
\]

where \( Z \) and \( Y \) are the random variables of the standard Gaussian and a Gamma distribution \( \Gamma(m/2, 2) \), respectively. A random variable of \( \Gamma(m/2, 2) \), in turn, can be obtained using an algorithm available, for example, in the literature [18].

A notable difference between the multivariate and the univariate Cauchy random variables lies in the “angular” parts of the distribution. Random vectors whose components are independently generated from a univariate Cauchy distribution are more concentrated along the coordinate axes, especially for large variations. On the contrary, random vectors generated from a multivariate Cauchy distribution are scattered over the variable space with angular
isotropy, including the inter-axes region, as Eq. (1) implied. Thus, the multivariate Cauchy random variable may have an advantage over the univariate one when variables in a given problem are correlated.

3. Multivariate FEP for Function Optimization

In this section, we describe FEP algorithm for the function optimization process using the mutation based on the multivariate Cauchy random variable. We implement the procedure of FEP according to the description used in Refs. [10], [11] with the self-adaptive mutation in ESs, following examples from the literature [2], [3]. In the optimization of a function \( f(\vec{x}) \) of \( n \) variables, the algorithm aims to find

\[
\vec{x}_{\min} \quad \text{such that} \quad \forall \vec{x}, \quad f(\vec{x}_{\min}) \leq f(\vec{x}).
\]

Here,

\[
\vec{x} = \{x(1), x(2), \ldots, x(n)\} \in S \quad \text{and} \quad S = \{x(j) \in R, L_j \leq x(j) \leq U_j \} \subseteq R^n,
\]

where \( L_j \) and \( U_j \) are the lower and upper limits of the \( j \)-th \((j = 1, 2, \ldots, n) \) variable, respectively.

FEP using MCM is similar to FEP using UCM and EP using GM, and can be described as follows:

Step 1: Generate the initial population consisting of \( \mu \) individuals, each of which can be represented by a set of real vectors \((\vec{x}_i, \vec{\eta}_i)\), \( i = 1, 2, \ldots, \mu \). Thus, each \( \vec{x}_i \) and \( \vec{\eta}_i \) have \( n \) variables

\[
\vec{x}_i = \{x_i(1), x_i(2), \ldots, x_i(n)\},
\]

\[
\vec{\eta}_i = \{\eta_i(1), \eta_i(2), \ldots, \eta_i(n)\}.
\]

Step 2: For MCM, generate a multivariate Cauchy random variable of \( n \) components \( M_j(0, 1) \) from the standard multivariate Cauchy distribution using the algorithm discussed in Sect. 2. For each parent \((\vec{x}_i, \vec{\eta}_i)\), create an offspring \((\vec{x}'_i, \vec{\eta}'_i)\) in the following manner. For \( j = 1, 2, \ldots, n \),

\[
\vec{x}'_j = x_j(j) + \eta_j(j) M_j(0, 1),
\]

\[
\vec{\eta}'_j = \eta_j(j) \exp\{\tau' N(0, 1) + \tau N_j(0, 1)\}
\]

where \( N(0, 1) \) denotes a standard Gaussian random variable fixed for each parent \( i \), and \( N_j(0, 1) \) represents newly generated standard Gaussian random variables for each variable \( j \). The parameter \( \tau \) and \( \tau' \) are commonly set to [2], [3]

\[
\tau = \frac{1}{\sqrt{2} \sqrt{\mu}} \quad \text{and} \quad \tau' = \frac{1}{\sqrt{2} \mu}.
\]

Step 3: From \( \mu \) parents and their \( \mu \) offspring, calculate their fitness values \( f_1, f_2, \ldots, f_{2\mu} \).

Step 4: Initialize winning functions for each parent and its offspring as \( w_i = 0 \), for \( i = 1, 2, \ldots, 2\mu \). For each \( i \), randomly select one fitness function, for example \( f_j \) \((j \neq i)\), and compare the two fitness functions. If \( f_i \) is less than \( f_j \), the winning function for the individual increases by one unit, i.e., \( w_i = w_i + 1 \). Perform this procedure \( q \) times for each parent and its offspring.

Step 5: For the winning function \( w_i \) \((i = 1, 2, \ldots, 2\mu)\), select \( \mu \) individuals that have the greater winning values to be the parents of the next generation.

Step 6: Repeat step 2 through step 5 until the required criteria are satisfied.

The conventional FEP and EP basically differ from FEP with MCM in that Eq. (10) in Step 2 is replaced by

\[
\vec{x}'_j = x_j(j) + \eta_j(j) C_j(0, 1),
\]

\[
\vec{x}'_j = x_j(j) + \eta_j(j) N_j(0, 1),
\]

where \( C_j(0, 1) \) and \( N_j(0, 1) \) respectively represent a newly generated standard univariate Cauchy and Gaussian random variables for each variable \( j \).

4. Experimental Results and Discussion

4.1 Selection of Test Functions

In order to investigate the effect of MCM, we consider the scalable functions in which the characteristics of the functions do not change when both the number of independent variables and their ranges vary. We classify scalable test functions into unimodal or multimodal. A unimodal function does not have local minima and has only one minimum, whereas a multimodal function has many local minima. Multimodal functions are further classified into separable, inseparable, and partially separable functions [19].

A function is called separable if there is no correlation among variables. Thus, a separable function can be decomposed into a linear combination of independent subfunctions of a single variable. This implies that the optimal value of each variable in the optimization of a separable function can be determined independently from all other variables. An inseparable function, on the other hand, contains correlations among variables. Thus, an inseparable function can not be optimized by finding the optimal value of each variable independently. A partially separable (or partially inseparable, for that matter) function consists of both separable and inseparable parts.

Table 1 lists the test functions along with their characteristics. These are benchmarking functions that were considered in earlier studies [10], [11], [20]. \( f_1 \) and \( f_2 \) are unimodal, i.e., none of them have local minima. \( f_3 \) and \( f_4 \) are separable in the sense that each function is composed of the sum over subfunctions of each variable. \( f_5 \) and \( f_6 \) are inseparable meaning that these functions contain correlations among variables. \( f_7 \) and \( f_8 \) are partially separable in the sense that these contain both separable and inseparable parts. The first part of both functions is separable, while the second part is inseparable. In addition, these functions contain trigonometric function as non-linear correlation.
Table 1 List of functions that were tested in this study. In the class column, ‘U’ and ‘M’ stand for the unimodal and the multimodal, respectively. In addition, ‘S’, ‘IS’, and ‘PS’ stand for the separable, inseparable, and partially separable, respectively.

<table>
<thead>
<tr>
<th>Test Functions</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 = \sum_{i=1}^{n} x_i^2 )</td>
<td>U, S</td>
</tr>
<tr>
<td>( f_2 = \sum_{i=1}^{n} (\sum_{j=1}^{n} x_i)^2 )</td>
<td>U, IS</td>
</tr>
<tr>
<td>( f_3 = -\sum_{i=1}^{n} x_i \sin(\sqrt{</td>
<td>v</td>
</tr>
<tr>
<td>( f_4 = \sum_{i=1}^{n} \left( x_i^2 - 10 \cos(2\pi x_i) + 10 \right) )</td>
<td>M, IS</td>
</tr>
<tr>
<td>( f_5 = 0.5 \left( \sum_{i=1}^{n} x_i^2 \right)^{0.5} \left( \sum_{i=1}^{n} x_i \right)^{-0.5} )</td>
<td>M, IS</td>
</tr>
<tr>
<td>( f_6 = \left( \sum_{i=1}^{n} x_i \right)^{0.25} \left( \sum_{i=1}^{n} x_i^2 \right)^{0.75} )</td>
<td>M, IS</td>
</tr>
<tr>
<td>( f_7 = -20 \exp \left( -0.2 \sqrt{\sum_{i=1}^{n} x_i^2} \right) - \sum_{i=1}^{n} \exp \left( \frac{1}{10} \cos(2\pi x_i) \right) + 20 + e )</td>
<td>M, PS</td>
</tr>
<tr>
<td>( f_8 = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} \cos \left( \frac{x_i}{\sqrt{n}} \right) + 1 )</td>
<td>M, PS</td>
</tr>
</tbody>
</table>

4.2 Experimental Setup and Statistical Hypothesis Test

We applied the algorithms discussed in Sect. 3 to the minimization of test functions in Table 1. We compared the performances of MCM versus UCM as well as MCM versus GM. The followings are the parameters used in the optimization, as suggested in earlier studies [2], [4], [10], [11]:

- population size: \( \mu = 100 \)
- tournament size: \( q = 10 \)
- initial mutation strength: \( \eta(j) = 3.0 \)
- number of generations: 5000

With the preset parameters, we varied both the number of variables and their ranges to illustrate the dependence and/or independence of the performance with respect to both factors.

Statistical hypothesis tests were carried out to discern the performance between the two sets of different mutation methods: MCM versus UCM and MCM versus GM. The performance is viewed in terms of the minimum value and the robustness. These measures are respectively quantified by the average and the standard deviation obtained from 50 independent runs. The smaller standard deviation a mutation has, the more robust the mutation is. For the statistical tests, the null hypothesis is set in such a way that the conventional mutations (UCM or GM) show better in minimized value or more robust than the proposed MCM. The obtained minimum value and the robustness are tested by using \( Z \) and \( F \) statistics, respectively. We tested the hypotheses with a 5% significance level (i.e., \( \alpha = 0.05 \)). Thus, the null hypothesis is rejected if \( Z > 1.65 \) for the minimum value test, and \( F > 1.61 \) for the robustness test.

4.3 Test Function Results and Discussion

4.3.1 Unimodal Functions

The results of the minimization of \( f_1 \) and \( f_2 \), together with the result of two statistical hypothesis tests, are shown in Table 2 for a number of variables and their ranges. Table 2 and Fig. 1 show that GM outperforms the other mutations. Moreover, MCM outperforms UCM.

Since \( f_1 \) is symmetric with respect to the angular part in the hyper-spherical polar coordinate, the optimization depends mainly on the radial part of the variates. A univariate Cauchy distribution generates longer jumps in the radial part than a multivariate Cauchy distribution due mainly to the independence among component. Furthermore, it is known that, due to an infinite variance, a Cauchy distribution can generate larger variation than a Gaussian distribution [10], [11]. Therefore, as shown in Fig. 1 (a), at early stages of evolution, UCM converges faster than both MCM and GM; furthermore MCM converges faster than GM. However, longer jumps may be detrimental when the population is located close to the minimum, and this occurs in the later stage of the evolution. Toward the later stage of the evolution, it is more likely to occur that the offspring generated by longer
Table 2  The averaged minimum values and their standard deviations (in parentheses) of functions $f_1$-$f_6$ are listed for a few selected numbers of variables $n$ and their ranges $[-R, R]$. The statistics are obtained by 50 independent runs. $Z_1$ and $F_1$ are test statistics between GM and MCM; $Z_2$ and $F_2$ are those between UCM and MCM. The values of test statistics with $^\dagger$ are significant at the level $\alpha = 0.05$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>$R$</th>
<th>Gaussian</th>
<th>univariate Cauchy</th>
<th>multivariate Cauchy</th>
<th>$Z_1$</th>
<th>$F_1$</th>
<th>$Z_2$</th>
<th>$F_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10.0</td>
<td>0.000018 (0.000004)</td>
<td>0.000004 (0.000012)</td>
<td>0.000030 (0.000008)</td>
<td>-4.4</td>
<td>0.26</td>
<td>3.94$^\dagger$</td>
<td>2.63$^\dagger$</td>
</tr>
<tr>
<td>10</td>
<td>50.0</td>
<td>0.000018 (0.000004)</td>
<td>0.000004 (0.000014)</td>
<td>0.000030 (0.000009)</td>
<td>-3.8</td>
<td>0.21</td>
<td>2.95$^\dagger$</td>
<td>2.47$^\dagger$</td>
</tr>
<tr>
<td>10</td>
<td>100.0</td>
<td>0.000018 (0.000005)</td>
<td>0.000004 (0.000012)</td>
<td>0.000031 (0.000008)</td>
<td>-4.3</td>
<td>0.36</td>
<td>3.73$^\dagger$</td>
<td>2.43$^\dagger$</td>
</tr>
<tr>
<td>20</td>
<td>10.0</td>
<td>0.000156 (0.000026)</td>
<td>0.000773 (0.000151)</td>
<td>0.000541 (0.000096)</td>
<td>-17.9</td>
<td>0.07</td>
<td>6.38$^\dagger$</td>
<td>1.86$^\dagger$</td>
</tr>
<tr>
<td>20</td>
<td>50.0</td>
<td>0.000153 (0.000022)</td>
<td>0.000745 (0.000154)</td>
<td>0.000554 (0.000090)</td>
<td>-19.2</td>
<td>0.06</td>
<td>4.81$^\dagger$</td>
<td>2.89$^\dagger$</td>
</tr>
<tr>
<td>20</td>
<td>100.0</td>
<td>0.000159 (0.000025)</td>
<td>0.000808 (0.000133)</td>
<td>0.000554 (0.000099)</td>
<td>-17.4</td>
<td>0.06</td>
<td>6.83$^\dagger$</td>
<td>1.82$^\dagger$</td>
</tr>
</tbody>
</table>

jumps tend to move away from the minimum. As a result, as generation proceeds, GM outperforms both UCM and MCM as can be seen from the test statistics of $Z_1$ and $F_1$ in Table 2. Within the Cauchy mutations, however, MCM outperforms UCM in that MCM obtains a better minimum value and is more robust than UCM as the test statistics of $Z_2$ and $F_2$ in Table 2 indicate. A similar trend has been demonstrated in Refs. [10], [11].

In addition to the unimodality, $f_2$ includes non-linear correlation among variables. Thus, MCM, which can more effectively search the interdependent region of variables, outperforms UCM for all stages of evolution, as shown in Fig. 1 (b) and the test statistics of $Z_2$ and $F_2$ in Table 2. In addition, although GM shows a slow convergence at an early stage of the evolution, it outperforms both UCM and MCM for the same reason as the case of $f_1$ as the test statistics of $Z_1$ and $F_1$ in Table 2 show.

4.3.2 Separable Functions

$f_3$ and $f_4$ are separable and multimodal functions in that each function is the sum of contributions from each variable and has many local minima. As illustrated in Table 2 in terms of obtained minimum values and the test statistics of $Z_1$ and $F_1$, both UCM and MCM outperform GM. This is consistent with the early finding [10], [11] in that a Cauchy distribution, because of its infinite variance, can investigate a wider area of the search space than a Gaussian distribution.

Between UCM and MCM, UCM outperforms MCM as the test statistics $Z_2$ in Table 2 shows. Given that the optimization of a separable function is obtained by independent optimization from each variable, UCM, which searches the variable space independently, is more suitable than MCM. As shown in Fig. 2, UCM outperforms MCM at all stages of evolution, as expected. In addition, the standard deviation of UCM is much smaller than that of MCM, meaning that
UCM is more robust than MCM as the test statistics $F_2$ in Table 2 supports.

### 4.3.3 Inseparable Functions

$f_5$ and $f_6$ are inseparable functions in that each of these functions is composed of non-linear correlations among variables. Due to the inseparability, the optimization of these functions cannot be obtained by the independent optimization of each variable.

As illustrated in Table 2, MCM outperforms both GM and UCM for all cases and the results are statistically significant in terms of the test statistics $Z_1$ and $Z_2$ for the obtained minimum values. The preference of MCM over UCM is due to the fact that the minimization of $f_5$ and $f_6$ depend on the correlation among variables. MCM can search interdependent regions of the variable more effectively than UCM. Moreover, Table 2 indicates that MCM is more suitable for a larger $n$. This is expected because as $n$ becomes large, the correlations among $n$ variables increase dramatically. In addition, the standard deviation of MCM is much smaller than that of UCM, meaning that MCM is more robust than UCM, especially when the number of variables $n$ is relatively small as the test statistics $F_1$ and $F_2$ demonstrate. This trend is more severe for the case of $f_6$. While MCM is much more robust than GM and UCM for small $n$, this characteristic no longer holds as $n$ increases.

As shown in Fig. 3, MCM outperforms both GM and UCM at all stages of evolution. In addition, the superiority of MCM and UCM to GM is again due to the search capability of a larger variable space. Although GM also takes into account correlations among variables, its relatively small variation of a variable limits an effective investigation of the search space. In summary, MCM is more effective and robust than both UCM and GM for the optimization of inseparable functions. Considering that, in general, the opti-
4.3.4 Partially Separable Functions

$f_7$ and $f_8$ are partially separable functions in the sense that the functions contain both separable and inseparable parts. More specifically, the first term of each function contains the sum over squares of variables as a separable part, whereas the second term includes non-linear trigonometric correlation among variables as an inseparable part. Moreover, the inseparable part includes the product of the trigonometric functions, which can takes values between $[-1, 1]$. Table 3 shows the results of the minimization of $f_7$ and $f_8$, together with statistical hypothesis tests, for a selected number of variables and their ranges.

As demonstrated in Table 3, the performance of MCM versus UCM depends on the range of variables. MCM tends to yield better result than UCM for relatively small ranges of the variables, whereas UCM produces better results for relatively large ranges. These results are supported by the fact that the test statistics $Z_2$ and $F_2$ in Table 3 are significant only for relatively small ranges of the variables. This characteristic can be understood as follows. Note that the separable part, the sum over the squares of variable, becomes smaller when the range of variable is small. On the contrary, the inseparable part, the product of trigonometric functions, always takes values between $[-1, 1]$ regardless of the range of the variable. This suggests that the inseparable part dominates at small ranges of variable, in which the separable part contributes little compared with the inseparable part. Therefore, we expect that MCM contributes more effectively when the range of variable is small, as shown, for instance, in Figs. 4 (a) and 5 (a). The situation, however, would be reversed when the range becomes large. In this case, the separable part becomes larger, while the trigonometric function still takes values between $[-1, 1]$. Consequently, the selection pressure is heavier on the separable part than on the inseparable part. Thus, as the range of the variable becomes large, the separable part is more important than the inseparable part in terms of the optimization.

Table 3  The averaged minimum values and their standard deviations (in parentheses) of functions $f_1$ and $f_8$ are listed for a few selected numbers of variables $n$ and their ranges $[-R,R]$. The statistics are obtained by $50$ independent runs. The captions for the test statistics are the same as those of Table 2.

<table>
<thead>
<tr>
<th></th>
<th>$n$</th>
<th>$R$</th>
<th>Gaussian</th>
<th>univariate Cauchy</th>
<th>multivariate Cauchy</th>
<th>$Z_1$</th>
<th>$F_1$</th>
<th>$Z_2$</th>
<th>$F_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>20</td>
<td>5.0</td>
<td>4.165665 (1.511189)</td>
<td>0.027366 (0.002891)</td>
<td>0.022265 (0.002119)</td>
<td>12.26†</td>
<td>508656†</td>
<td>6.36†</td>
<td>1.86†</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>10.0</td>
<td>7.287442 (3.054462)</td>
<td>0.027104 (0.002385)</td>
<td>0.022481 (0.001608)</td>
<td>10.64†</td>
<td>3607162†</td>
<td>7.19†</td>
<td>2.20†</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>15.0</td>
<td>15.240721 (2.343411)</td>
<td>0.027074 (0.002411)</td>
<td>0.022362 (0.002138)</td>
<td>29.04†</td>
<td>120090†</td>
<td>6.54†</td>
<td>1.27†</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>30.0</td>
<td>18.052600 (1.827508)</td>
<td>0.027207 (0.002366)</td>
<td>1.561453 (4.750187)</td>
<td>14.49†</td>
<td>0.15</td>
<td>-1.44</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>40.0</td>
<td>19.124115 (2.418141)</td>
<td>0.026893 (0.002340)</td>
<td>15.92913 (7.356057)</td>
<td>2.08†</td>
<td>0.11</td>
<td>-9.43</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>50.0</td>
<td>19.695509 (1.95653)</td>
<td>0.426919 (2.814357)</td>
<td>18.880868 (3.923998)</td>
<td>0.89</td>
<td>0.09</td>
<td>-17.09</td>
<td>0.51</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>60.0</td>
<td>18.76522 (0.663287)</td>
<td>6.397059 (9.366645)</td>
<td>19.53972 (2.813828)</td>
<td>0.52</td>
<td>0.06</td>
<td>-6.01</td>
<td>11.05†</td>
</tr>
<tr>
<td>$f_8$</td>
<td>30</td>
<td>2.0</td>
<td>2.403361 (0.546029)</td>
<td>0.048943 (0.003696)</td>
<td>0.042548 (0.004036)</td>
<td>23.68†</td>
<td>18297†</td>
<td>6.40†</td>
<td>0.84</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>5.0</td>
<td>4.773656 (1.268550)</td>
<td>0.050728 (0.003954)</td>
<td>0.044465 (0.004738)</td>
<td>20.42†</td>
<td>71683†</td>
<td>5.56†</td>
<td>0.70†</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>10.0</td>
<td>9.034392 (2.039632)</td>
<td>0.050942 (0.004265)</td>
<td>0.048572 (0.014382)</td>
<td>24.13†</td>
<td>20111†</td>
<td>0.87†</td>
<td>0.09†</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>20.0</td>
<td>16.000835 (2.521438)</td>
<td>0.049856 (0.003385)</td>
<td>0.891795 (3.404077)</td>
<td>19.54†</td>
<td>0.55</td>
<td>-1.35</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>30.0</td>
<td>18.496764 (1.275932)</td>
<td>0.051261 (0.003556)</td>
<td>11.962069 (5.85124)</td>
<td>4.12†</td>
<td>0.02</td>
<td>-7.60</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>40.0</td>
<td>19.527476 (0.392722)</td>
<td>0.053613 (0.003331)</td>
<td>17.840579 (5.307685)</td>
<td>1.74†</td>
<td>0.01</td>
<td>-18.36</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Fig. 4  Plots of the minimum function $f_7$ versus generations for MCM (solid line), UCM (dotted line), and GM (dashed line). The results are averaged over 50 independent runs with $n = 20$ and the range $[-5.0, 5.0]$ (a) and $[-60.0, 60.0]$ (b).
Therefore, we expect that UCM yields better results than MCM when the range of variable is large, as shown, for example, in Figs. 4 (b) and 5 (b). From the above findings, we can infer that, in the optimization, MCM outperforms UCM when the inseparable part dominates, and a reverse result is obtained when the separable term dominates. In addition, the robustness of the two different mutations is more or less, if not totally, correlated with the optimized values.

The performance of MCM versus GM at relatively small ranges of variable depends on the characteristic of the test function. As shown in Table 3 and Figs. 4 (a) and 5 (a), MCM outperforms GM for \( f_7 \); however, GM shows better performance for \( f_8 \). Since both MCM and GM can reflect correlations among variables, the separability or inseparability is not an important issue in this case. There is no clear explanation for this characteristic at this point, and it deserves further investigation. In contrast, at relatively large ranges of variable, MCM outperforms GM for both \( f_7 \) and \( f_8 \) as shown, for instance, in Figs. 4 (b), 5 (b), and the test statistics of \( Z_1 \) and \( F_1 \) in Table 3. This is due to the search capability of MCM for a wider area of the variable space than GM.

5. Summary and Conclusion

In this study, we proposed a mutation operation in FEP based on the multivariate Cauchy probability distribution and investigated the advantage of the proposed mutation in conjunction with the conventional univariate Cauchy and Gaussian mutations. To this end, we introduced a method of generating a random vector from a multivariate Cauchy distribution and analyzed the characteristics of the multivariate Cauchy mutation in comparison with the conventional univariate Cauchy and Gaussian mutations.

We showed that the multivariate Cauchy mutation could search interdependent regions in the variable space due to the dependence among the components of the multivariate random variable. This suggested that the multivariate Cauchy mutation was effective for an optimization process in which correlations exist among variables. In addition, the multivariate Cauchy mutation maintained the search ability of the univariate Cauchy mutation for a wider area of the variable space compared with the Gaussian mutation. This implies that the multivariate Cauchy mutation could search the inter-axes region of variable space while maintaining the search capability of the wider variable space. Thus, the multivariate Cauchy mutation possessed the advantages over both the univariate Cauchy mutation and the Gaussian mutation.

We applied the multivariate Cauchy mutation to various types of test functions: unimodal, separable, inseparable, and partially separable functions. We found that the multivariate Cauchy mutation outperformed for the case of inseparable functions and partially separable functions in which the inseparable parts are dominated. With these, we could argue that the multivariate Cauchy mutation is useful in optimizing functions having correlation among variables.

Considering that the optimization of correlated variables is, in general, more interesting than uncorrelated variables, the multivariate Cauchy mutation may serve a better strategy for the optimization algorithm. Although one cannot claim that the multivariate Cauchy mutation always produces better results for all optimization problems, there is certainly an advantage to the multivariate Cauchy mutation over the univariate Cauchy and the Gaussian mutations.

This study is the first step toward an investigation of the role of the multivariate Cauchy distribution in FEP. Further studies regarding not only the algorithm itself, but practical optimization problems, will be valuable and support the advantage of the multivariate Cauchy mutation. In addition, any variation of the proposed method would be interesting and merit further investigation. This would bring a more complete adaptive algorithm to FEP and, perhaps, a direction for future study.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2010-0022163).

References


Chang-Yong Lee received BS degree in computer science and statistics from Seoul National University, South Korea in 1983, and Ph.D degree from the University of Texas at Austin in 1995 for research in computational and nuclear physics. From 1996 to 1998, he was a Senior Researcher at the Basic Research Center, Electronics and Telecommunication Research Institute in South Korea. In 1998, he joined Kongju National University in South Korea as an Assistant Professor, where currently he is a Professor in the Department of Industrial and Systems Engineering. His main research interests include the evolutionary computations, Internet ecology, and bioinformatics. Dr. Lee has his biographical profile included in the 7th edition of Marquis Who’s Who in Science and Engineering.

Yong-Jin Park received BS and MS degrees of Crop Science from Seoul National University, South Korea in 1987 and 1989, respectively. He also received Ph.D degree from Seoul National University in 1997 for the research in Plant Genetics and Breeding. From 1997 to 2005, he was a senior scientist at the department of Researches on Molecular Markers and Crop Evolution, National Institute of Agricultural Biotechnology in South Korea. During 2005–2007, he served as the Project Leader of Medicinal Plants and Molecular Characterization, Regional Office for Asia, the Pacific and Oceania (APO), Bioversity International in Malaysia. Since 2009, he is a professor at the department of Plant Resources, Kongju National University in South Korea. His research interests includes Comparative Genomics, Molecular Genetics on Plant Genetic Resources (PGRs).