

MP-EDA: A Robust Estimation of Distribution Algorithm with Multiple Probabilistic Models for Global Continuous Optimization^{*}

Jing-hui Zhong¹, Jun Zhang^{1,**}, and Zhun Fan²

¹ Dept. of Computer Science, SUN yat-sen University
Key Laboratory of Digital Life (Sun Yat-Sen University), Ministry of Education
Key Laboratory of Software Technology, Education Department of Guangdong
Province, P.R. China
junzhang@ieee.org

² Denmark Technical University

Abstract. Extending Estimation of distribution algorithms (EDAs) to the continuous field is a promising and challenging task. With a single probabilistic model, most existing continuous EDAs usually suffer from the local stagnation or a low convergence speed. This paper presents an enhanced continuous EDA with multiple probabilistic models (MP-EDA). In the MP-EDA, the population is divided into two subpopulations. The one involved by histogram model is used to roughly capture the global optima, whereas the other involved by Gaussian model is aimed at finding highly accurate solutions. During the evolution, a migration operation is periodically carried out to exchange some best individuals of the two subpopulations. Besides, the MP-EDA adaptively adjusts the offspring size of each subpopulation to improve the searching efficiency. The effectiveness of the MP-EDA is investigated by testing ten benchmark functions. Compared with several state-of-the-art evolutionary computations, the proposed algorithm can obtain better results in most test cases.

Keywords: Estimation of Distribution Algorithm, Evolutionary Computation, Histogram, Multivariate Gaussian Distribution, Global Optimization.

1 Introduction

The estimation of distribution algorithms (EDAs) are a new class of evolutionary computation algorithms [1] [2]. They generate new individuals by sampling a probabilistic model, which is estimated based on the current promising solutions. As the probabilistic model can capture promising areas in a statistically sound manner and can explicitly express the interactions among variables, EDAs usually can outperform traditional EAs on a number of complex problems .

^{*} This work was supported in part by the National Natural Science Foundation of China No. U0835002 and No.61070004, by the National High-Technology Research and Development Program (“863” Program) of China No. 2009AA01Z208.

^{**} Corresponding author.

The EDAs are first proposed to solve discrete problems with binary representation. For the last few years, various efforts have been made on extending them to the continuous optimization [3] - [11]. Most existing continuous EDAs use the Gaussian model or the histogram model to estimate the distribution of promising areas. The marginal Gaussian probabilistic model was used to guide the search in early continuous EDAs, such as the PBILc [3] and the UMDAc [4]. Lately, the multivariate Gaussian models also appeared in continuous EDAs [5] [6]. EDAs with a single Gaussian model are excellent in finding the global optima for unimodal optimization. However, for multimodal problems, they usually suffer from a slow convergence speed or even a local stagnation. Though clustering techniques have been utilized to address some of these issues [7], more efficient methods are greatly desirable.

Meanwhile, the histogram has multimodal density and can capture multiple local optima at the same time. Hence the histogram-based EDAs (HEDAs) are less likely to get trapped in local optima [8]. However, the HEDAs usually need a heavy computational cost to search for a highly accurate solution. To tackle this problem, the local search techniques [9], the shrink strategy [10], and the sub-divided method [11], have been proposed in recent few years. Nevertheless, since the histogram ignores interactions among variables, the HEDAs are not efficient enough to solve problems containing variable dependency.

In this paper, we present an enhanced EDA with multiple probabilistic models (named MP-EDA) for the global continuous optimization. The proposed MP-EDA has made two major improvements as below. First, the robustness to find a highly accurate solution is guaranteed by adopting two different types of probabilistic models: the multivariate Gaussian model and the fixed-height histogram (FHH) model. Second, the efficiency of the search is improved by utilizing an adaptive control strategy. In the MP-EDA, the population is initially divided into two subpopulations, with each involved by a probabilistic model. The one involved by the FHH is used to roughly capture the global optima, and the other involved by the multivariate Gaussian model is aimed at finding highly accurate solutions. During the evolution, a migration operation is carried out periodically to exchange some best individuals between the two subpopulations. By sharing information of better individuals, both subpopulations can converge faster. Besides, in order to improve the search efficiency, an adaptive strategy is used to adjust the offspring sizes of the two subpopulations. The performance of a probabilistic model is measured by the average fitness of new individuals generated by it during recent generations. The better probabilistic model is allowed to generate more offspring individuals, and vice versa. We investigate the effectiveness of the proposed MP-EDA by solving ten test functions with different characteristics. Several state-of-the-art EAs (i.e. PLSO [12], DE [13], FEP [14] and CMA-ES [15]) are used for the comparison. Experimental results show that the proposed MP-EDA can achieve better results in most test cases.

The rest of the paper is organized as follows. Section 2 briefly describes the EDAs' general framework and two classical continuous EDAs. Section 3 illustrates the detailed implementations of the MP-EDA. The experimental studies on the MP-EDA are presented in Section 4. At last, Section 5 draws the conclusions.

2 Estimation of Distribution Algorithms

2.1 Framework of EDAs

The general framework of EDAs is similar to that of the GA. However, there is neither crossover nor mutation in EDAs. Instead, they use a probabilistic model to generate new individuals. Specifically, the common outline of EDAs consists of the following four steps.

Step 1: Initialize the algorithm parameters and the initial population.

Step 2: Select a certain number of excellent individuals. There are several selection strategies, such as the truncation selection and the tournament selection.

Step 3: Construct probabilistic model by analyzing information of the selected individuals.

Step 4: Create new population by sampling new individuals from the constructed probabilistic model.

There is a repetition from *Step2* to *Step4* until the algorithm meets the termination condition.

2.2 Multivariate Gaussian Model

The multivariate Gaussian model is most commonly used in continuous EDAs. It characterizes the distribution of data points by two parameters μ and Σ . The former gives the mean vector of data points, while the latter describes the covariance information. In particular, given a set of data points $X = \{x_1, x_2, \dots, x_Z\}$, the multivariate Gaussian model of X can be expressed by

$$N(\mu, \Sigma) = \frac{1}{\sqrt{2\pi} \Sigma} \cdot \exp\left(-\frac{1}{2} \cdot (x - \mu) \cdot \Sigma^{-1} \cdot (x - \mu)\right) \quad (1)$$

where μ is the mean vector of X , and Σ is the covariance matrix of X .

The multivariate Gaussian model is able to capture the interaction between variables, owing to the covariance matrix in the density function. However, it is not effective to estimate the distribution of multimodal data points, due to its unimodal density.

2.3 Histogram Model

There are usually two types of histograms, namely the fixed-width histogram (FWH) and the FHH. The FWH consists of bins with the same width, whereas the FHH contains bins with the same height. In this paper, we choose the FHH as one probabilistic model of the proposed method, because it has shown to perform better than the FWH. Given a set of data points $X = \{x_1, x_2, \dots, x_Z\}$ within an interval $\Omega = [x_{\min}, x_{\max}]$. The FHH of X consists of n bins, with each bin B_i ($i=1, 2, \dots, n$) containing the same number of data points. Hypothesizing that the data points in each bin are uniformly

distributed, the estimated density $\hat{f}_H(x)$ of the underlying probability density at any point x can be computed by

$$\hat{f}_H(x) = \frac{1}{w_k \cdot n} \quad (2)$$

where w_k is the width of the bin B_k that contains x . According to the definition of the FHH, bins in the dense regions would have a narrower width than those in the sparse regions. As the population of an evolutionary algorithm evolves, individuals will gradually gather around promising areas. By giving more bins to these regions, the FHH is able to capture multiple peaks at the same time. Nevertheless, one drawback of the FHH based EDA is that it may get stuck when the population converges. In this case, a heavy computational cost is usually required to sample a highly accurate solution.

3 MP-EDA for Multimodal Continuous Optimization

3.1 Algorithm Framework

There are two subpopulations (POP_{fhh} and POP_{gm}) involved by two probabilistic models in the MP-EDA. The POP_{fhh} is involved by the FHH model, whereas the POP_{gm} is involved by the Gaussian model. Specifically, the framework of the proposed MP-EDA contains following six steps.

1) Step 1 - Initialization

This step initializes parameters of the algorithm, such as the size of the population N , the number of excellent individuals S , the migration cycle T and the number of bins n . Besides, new individuals of POP_{fhh} and POP_{gm} are randomly generated in the search space, with their offspring sizes respectively set as $N_{fhh} = N$ and $N_{gm} = N$.

2) Step 2 - Constructing probabilistic models

This step aims at constructing two different probabilistic models for the two subpopulations in parallel. For the POP_{fhh} , S best individuals are firstly selected. Then the marginal histogram for all variables is updated. Let $X = \{x_0, x_1, \dots, x_S\}$ with $x_0 \leq x_1 \leq \dots \leq x_S$, be the i -th variable values of these selected individuals, then the lower bound and the upper bound of each bins is updated by

$$l_{j,i} = \begin{cases} \text{lower bound of the } i\text{-th variable,} & \text{if } j = 1 \\ (x_{\lfloor (j-1) \cdot \Delta x \rfloor} + x_{\lfloor (j-1) \cdot \Delta x \rfloor + 1}) / 2, & \text{otherwise} \end{cases} \quad (3)$$

$$u_{j,i} = \begin{cases} \text{upper bound of the } i\text{-th variable,} & \text{if } j = n \\ l_{j+1,i}, & \text{otherwise} \end{cases} \quad (4)$$

where $\Delta x = S / n$ is the average number of data points in each bin.

As for the multivariate Gaussian model, the constructing process aims to update values of two parameters: the mean μ and the covariance matrix Σ . Firstly, S promising

individuals are selected in POP_{gm} , and then the mean and the covariance of these S individuals are computed and set as the values of μ and Σ .

3) Step 3 – Sampling new population

The sampling process of the FHH consists of two sub-steps. Firstly, a bin is randomly selected according to the constructed marginal histogram. Then a random value uniformly distributed in the interval of the selected bin is generated.

As for the multivariate Gaussian model, the sampling process contains three sub-steps.

Sub-step1: Use the Cholesky decomposition to generate a lower triangular matrix S , with

$$\Sigma = S \cdot S^T \quad (5)$$

Sub-step2: obtain a 1-by- D matrix Z with random elements sampled from a standard Normal distribution $N(0,1)$.

Sub-step3: generate a individual P by

$$P = \mu + S \cdot Z \quad (6)$$

In order to maintain higher population diversity, a scale parameter $\delta > 1$ is suggested to add in (6) [16] and expressed as

$$P = \mu + \delta \cdot S \cdot Z \quad (7)$$

Note that the parameter δ should be set appropriate, for a trade-off between the diversity and the convergence of the population.

4) Step 4 –Evaluation and Replacement

This step aims to evaluate the fitness values of all new individuals and replace some worst individuals of the current population by better new individuals. For example, let $C = \{C_1, C_2, \dots, C_N\}$ be the current subpopulation POP_{fhh} , $C^* = \{C^*_1, C^*_2, \dots, C^*_N\}$ be the newly generated individuals by the FHH and $C^\# = C \cup C^*$. Then the best N individuals in $C^\#$ are selected as the new POP_{fhh} . The similar process can be done to generate the new POP_{gm} .

5) Step 5 –Migration

During the evolution, a migration operation is carried out every T generations. In the migration, the best M individuals in one subpopulation are migrated to the other. By sharing the best individuals of two subpopulations, the algorithm can utilize the advantages of both probabilistic models, which makes the algorithm less likely to be trapped into local optima and more efficient to find highly accurate solutions.

6) Step 6 – Adaptive control operation

This step aims to adjust the size of new offspring for each subpopulation, so as to save the computational cost. The key idea is to make the better probabilistic model generate more new individuals in each generation. Specifically, F_{fhh} and F_{gm} are supposed to be the average fitness of new individuals generated by the FHH and the GM respectively during the recent T generations. If F_{fhh} is better than F_{gm} , the value of N_{fhh} would be enlarged by a small step Δ , whereas the value of N_{gm} would be reduced

by Δ . Otherwise, the reverse operation can be done. The maximum and the minimum of N_{fhh} and N_{gm} are both set as P_{min} and P_{max} .

There is a repetition from *Step2* to *Step6* and the evolution processes iteratively until reaching the maximum number of evaluations.

4 Experiments and Comparisons

4.1 Test Functions and Parameter Settings

Ten benchmark functions are contained in the experimental study, as listed in Table 1. These functions are chosen in the literature [17] and [18]. Parameter settings of all comparison algorithms are listed in Table 2. The parameter settings of the DE, the CLPSO and the CMA-ES are set as those in the reference papers. The dimension of all test functions is 30 and the maximum evaluation number for each function is 3×10^5 . In order to make fair comparisons, 30 independent runs are performed on each algorithm.

Table 1. Test functions

Test Function		Domain
Unimodal	$f_1(x) = \sum_{i=1}^n z_i^2 + f_bias, z = x - o$	[-100,100]
	$f_2(x) = \sum_{i=1}^n (\sum_{j=1}^i z_j^2) + f_bias, z = x - o$	[-100,100]
	$f_3(x) = (\sum_{i=1}^n (\sum_{j=1}^i z_j^2)) \cdot (1 + 0.4 N(0,1)) + f_bias, z = x - o$	[-100,100]
Multimodal	$f_4(x) = \sum_{i=1}^n (100(z_i^2 - z_{i+1})^2 + (z_i - 1)^2) + f_bias, z = x - o$	[-100,100]
	$f_5(x) = \sum_{i=1}^n (-z_i \sin(\sqrt{ z_i }) + f_bias, z = x - o$	[-500,500]
	$f_6(x) = \sum_{i=1}^n (z_i^2 - 10\cos(2\pi z_i) + 10) + f_bias, z = x - o$	[-5,5]
	$f_7(x) = \sum_{j=1}^n \sum_{i=1}^n (\frac{y_{i,j}^2}{4000} - \cos(y_{i,j}) + 1) + f_bias,$ where $y_{i,j} = 100(z_j - z_i^2)^2 + (1 - z_i)^2, z = x - o$	[-100,100]
	$f_8(x) = \sum_{i=1}^n (\frac{z_i^2}{4000}) - \prod_{i=1}^n \cos(\frac{z_i}{\sqrt{i}}) + 1 + f_bias, z = x - o$	[-600,600]
	$f_9(x) = -20 \cdot \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n z_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi z_i) +$ $20 + e + f_bias, z = x - o$	[-32,32]
	$f_{10}(x) = \sum_{i=1}^n (z_i^2 - 10\cos(2\pi z_i) + 10) + f_bias, z = (x - o) * M$	[-5,5]

Table 2. Parameter Settings

Algorithm	Parameter Settings
GM-EDA	population size $N = 200$, $S=100$, $\alpha=0.2$, $Q=2$, $\delta=1.2$
FHH-EDA	population size $N = 200$, $S=100$, $n=50$
MP-EDA	population size $T=5$, $N_{\min}=20$, $N_{\max}=200$, $\Delta=10$, $M=5$
CLPSO	particle number $ps = 40$, $w_0 = 0.9$, $w_1 = 0.4$, $c = 1.49445$, $m = 7$
DE	population size = 100, $F = 0.5$, $CR = 0.9$
CMA-ES	population size = 14 as default

Table 3. Experimental Results. ‘ a ’ stands for the average best values, ‘ s ’ stands for the standard deviations and ‘ r ’ stands for the performance ranking of the corresponding algorithm.

			MP-EDA	FHH-EDA	GM-EDA	CLPSO	DE	CMA-ES
Unimodal	f_1	a	0	6.99×10^{-1}	2.08×10^{-14}	5.68×10^{-14}	0	3.79×10^{-15}
		s	0	3.37×10^{-1}	2.74×10^{-14}	0	0	1.44×10^{-14}
		r	1	6	4	5	1	3
	f_2	a	1.98×10^{-9}	3.40×10^3	3.51×10^{-10}	6.21×10^2	4.81×10^{-5}	3.22×10^{-14}
		s	2.23×10^{-9}	9.65×10^3	2.91×10^{-10}	1.52×10^2	4.00×10^{-5}	2.86×10^{-14}
		r	3	6	2	5	4	1
	f_3	a	2.46×10^{-5}	5.22×10^3	1.40×10^{-6}	6.29×10^3	1.39×10^{-2}	2.69×10^4
		s	4.56×10^{-5}	1.53×10^3	5.21×10^{-6}	1.38×10^3	1.38×10^{-2}	8.53×10^3
		r	2	4	1	5	3	6
Multimodal	f_4	a	4.74×10^{-5}	2.97×10^3	3.46×10^{-8}	5.14×10^0	2.87×10^0	5.12×10^{-14}
		s	1.03×10^{-4}	3.82×10^3	9.21×10^{-8}	4.62×10^0	1.98×10^0	1.73×10^{-14}
		r	3	6	2	5	4	1
	f_5	a	-12569.5	-12569.3	-6.03×10^3	-12557.6	-8290.5	-9555.6
		s	1.82×10^{-12}	6.75×10^{-2}	1.11×10^3	3.55×10^1	8.32×10^2	4.40×10^2
		r	1	2	6	3	5	4
	f_6	a	0	2.72×10^{-1}	1.85×10^2	3.69×10^{-13}	1.28×10^2	2.81×10^2
		s	0	1.79×10^{-1}	9.27×10^0	3.55×10^{-13}	2.08×10^1	3.27×10^1
		r	1	3	5	2	4	6
	f_7	a	3.51×10^2	5.46×10^2	6.79×10^2	2.53×10^2	4.95×10^2	6.88×10^1
		s	3.06×10^1	1.11×10^2	1.61×10^1	5.29×10^1	5.98×10^1	2.44×10^1
			3	5	6	2	4	1
	f_8	a	0	4.69×10^{-1}	1.52×10^{-14}	5.68×10^{-14}	0	2.47×10^{-2}
		s	0	1.67×10^{-1}	2.51×10^{-14}	0	0	0
		r	1	6	3	4	1	5
	f_9	a	5.68×10^{-14}	1.99×10^{-1}	1.06×10^{-13}	1.27×10^{-13}	6.06×10^{-14}	1.63×10^1
		s	0	7.99×10^{-2}	1.93×10^{-14}	2.40×10^{-14}	1.42×10^{-14}	7.41×10^0
		r	1	5	3	4	2	6
	f_{10}	a	4.13×10^1	1.58×10^2	1.90×10^2	1.03×10^2	1.78×10^2	5.13×10^2
		s	1.13×10^1	7.51×10^1	8.08×10^0	1.68×10^1	9.69×10^1	2.36×10^2
		r	1	3	5	2	4	6
Total r		17	46	37	37	32	39	
Average r		1.7	4.6	3.7	3.7	3.2	3.9	

4.2 Experimental Results

The comparison results are summarized in Table 3. Firstly, we investigate the performance of the FHH-EDA, GM-EDA and the MP-EDA. Results of f_1 show that, among these three algorithms, the MP-EDA performs the best, while the FHH-EDA performs the worst. The GM-EDA can obtain very highly accurate solutions, but it gets stuck near the global optima. This is because the eigenvectors of the covariate matrix in the GM-EDA has dropped to zero when the mean vector was still a bit distant from the global optimum. However, the MP-EDA can overcome this drawback, owing to the diversity provided by the histogram. Results of f_2 and f_3 indicate that, the FHH-EDA is not suitable to solve these kinds of problems, since it contains no mechanism to deal with variable interactions. However, the GM-EDA is very effective to solve these problems, and the proposed MP-EDA can also give competitive performance. As for multimodal functions, the MP-EDA performs much better than the GM-EDA and the FHH-EDA in all test cases except for the shift rosenbrock function. It can be observed that the Gaussian-based EDA can find satisfying solutions for multimodal functions with a big-valley structure (such as f_8 and f_9), but it performs

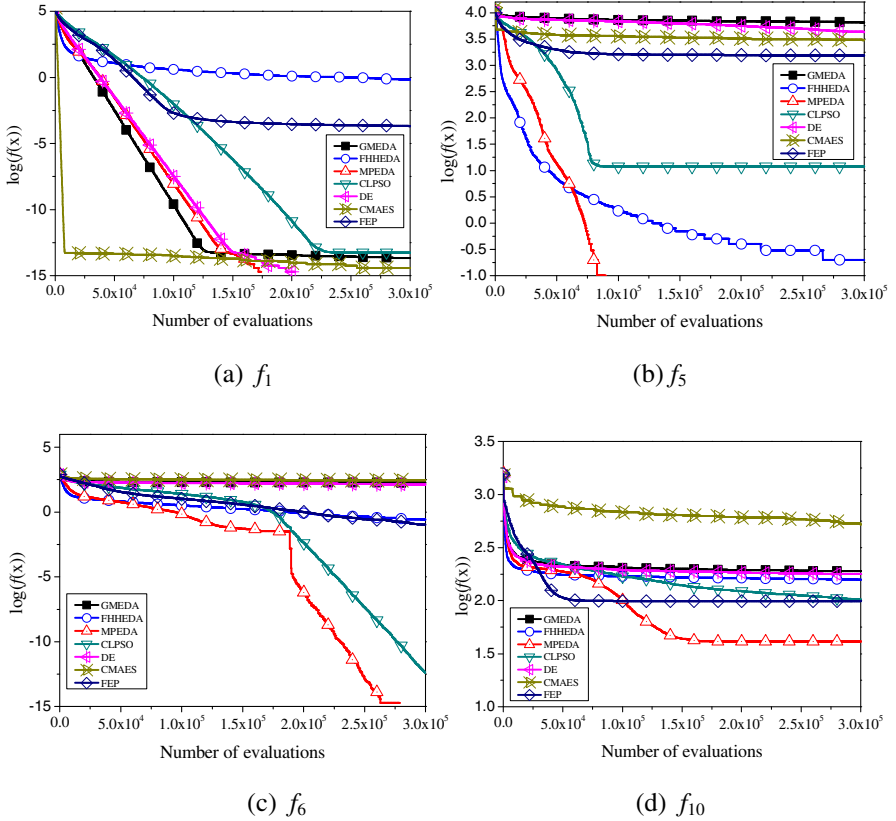


Fig. 1. Convergence graphs of the MP-EDA

very badly for those with local optima scattering in the search space. Meanwhile, the FHH-EDA can only find roughly accurate solutions for separable-multimodal functions (such as f5 and f6), but it performs badly for those containing variable interactions (such as f4). By combining two different probabilistic models, the MP-EDA performs better and more stable.

Results in table 3 indicate that the proposed MP-EDA performs the best in six test functions, and it can also give competitive performance on other test functions. According to the average rank value, the MP-EDA performs averagely much better than the other comparison EAs.

The convergence graphs of the MP-EDA and the comparison algorithms are shown in Fig. 1. It can be observed that the proposed algorithm can not only search a highly accurate solution, but also has a very fast convergence speed. Besides, the proposed MP-EDA can give good performance on both unimodal functions and multimodal functions.

5 Conclusions

This paper has proposed an enhanced EDA with multiple probabilistic models for the global continuous optimization. The proposed MP-EDA adopts a histogram and a multivariate Gaussian model to involve two subpopulations. During the evolution, a migration strategy is used to exchange some best individuals of the two subpopulations. Besides, the offspring size of each subpopulation is adaptively adjusted to reduce the computational cost. The effectiveness of the proposed MP-EDA has been investigated by testing ten benchmark functions. Compared with several state-of-the-art evolutionary algorithms, the proposed MP-EDA can obtain better results in most test cases.

This study has shown that, EDAs with multiple probabilistic models usually can work more effectively and efficiently than those with a single probabilistic model. As for future work, we will further extent the framework by hybridizing more probabilistic models. Besides, applying the proposed algorithm framework to solve real application is another promising research topic.

References

1. Baluja, S.: Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning. Technical report CMU-CS-94-163, Carnegie Mellon University (1994)
2. Mühlenbein, H., Paaß, G.: From recombination of genes to the estimation of distributions I. binary parameters. In: Ebeling, W., Rechenberg, I., Voigt, H.-M., Schwefel, H.-P. (eds.) PPSN 1996. LNCS, vol. 1141, pp. 178–187. Springer, Heidelberg (1996)
3. Sebag, M., Ducoulombier, A.: Extending population-based incremental learning to continuous search spaces. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) PPSN 1998. LNCS, vol. 1498, pp. 418–427. Springer, Heidelberg (1998)
4. Larrañaga, P., Etxeberria, R., Lozano, J.A., Peña, J.M.: Optimization in continuous domains by learning and simulation of Gaussian networks. In: Proceedings of the Genetic and Evolutionary Computation Conference 2000, Las Vegas, Nevada (July 2000)

5. Paul, T.K., Iba, H.: Real-Coded Estimation of Distribution Algorithm. In: Proceedings of The Fifth Metaheuristics International Conference (2003)
6. Yuan, B., Gallagher, M.: Experimental results for the special session on real-parameter optimization at CEC 2005: a simple, continuous EDA. In: Proc. of Congress on Evolutionary Computation (CEC 2005), vol. 2, pp. 1792–1799 (2005)
7. Lu, Q., Yao, X.: Clustering and learning Gaussian distribution for continuous optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 35(2), 195–204 (2005)
8. Tsutsui, S., Pelikan, M., Goldberg, D.E.: Evolutionary algorithm using marginal histogram models in continuous domain. *IlliGAL Report No. 2001019, UIUC* (2001)
9. Zhang, Q., Sun, J., Tsang, E., Ford, J.: Hybrid Estimation of Distribution Algorithm for Global Optimisation. *Engineering Computations* 21(1), 91–107 (2004)
10. Ding, N., Zhou, S., Sun, Z.: Histogram-based estimation of distribution algorithm: a competent method for continuous optimization. *Computer Science and Technology* 23(1), 35–42 (2008)
11. Xiao, J., Yan, Y.P., Zhang, J.: HPBILc: A histogram-based EDA for continuous optimization. *Applied Mathematics and Computation* 215(3), 973–982 (2009)
12. Liang, J.J., Qin, A.K., Suganthan, P.N., Baskar, S.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. on Evolutionary Computation* 10(3), 280–295 (2006)
13. Storn, R.M., Price, K.V.: Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optimization* 11, 341–359 (1997)
14. Yao, X., Liu, Y., Lin, G.: Evolutionary programming made faster. *IEEE Trans. On Evolutionary Computation* 3(2), 82–102 (1999)
15. Auger, A., Hansen, N.: Performance evaluation of an advanced local search evolutionary algorithm. In: Proc. of IEEE Congress on Evolutionary Computation (CEC 2005), vol. 2, pp. 1777–1784 (September 2005)
16. yuan, B., Gallagher, M.: On the importance of diversity maintenance in estimation of distribution algorithms. In: Proc. of the Genetic and Evolutionary Computation Conference-GECCO-2005, pp. 719–726. ACM, New York (2005)
17. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y.-P., Auger, A., Tiwari, S.: Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization. Nanyang Technol. Univ., Singapore, IIT Kanpur, India, KanGAL Rep. 2005005 (May 2005)
18. Noman, N., Iba, H.: Accelerating differential evolution using an adaptive local search. *IEEE Transactions on Evolutionary Computation* 12(1), 107–125 (2008)