

# Optimizing Continuous Problems Using Estimation of Distribution Algorithm Based on Histogram Model<sup>\*</sup>

Nan Ding<sup>1</sup>, Shude Zhou<sup>2</sup>, and Zengqi Sun<sup>2</sup>

<sup>1</sup> Department of Electronic Engineering, Tsinghua University, Beijing, China  
ding-n04@mails.tsinghua.edu.cn

<sup>2</sup> Department of Computer Science and Technology, Tsinghua University, Beijing, China  
zsd03@mails.tsinghua.edu.cn, szq-dcs@tsinghua.edu.cn

**Abstract.** In the field of estimation of distribution algorithms, choosing probabilistic model for optimizing continuous problems is still a challenging task. This paper proposes an improved estimation of distribution algorithm (HEDA) based on histogram probabilistic model. By utilizing both historical and current population information, a novel learning method – accumulation strategy – is introduced to update the histogram model. In the sampling phase, mutation strategy is used to increase the diversity of population. In solving some well-known hard continuous problems, experimental results support that HEDA behaves much better than the conventional histogram-based implementation both in convergence speed and scalability. Compared with UMDA-Gaussian, SGA and CMA-ES, the proposed algorithms exhibit excellent performance in the test functions.

## 1 Introduction

Recently, estimation of distribution algorithms have become the hot topic in the field of evolutionary computation [1]. The contribution of EDAs not only lies in its ability to explicitly learn the linkage relationship among variables, but also provides a novel macroscopical evolutionary paradigm, in which without any conventional operators, the population evolves by iteratively learning and sampling the probabilistic distribution model that describes the movements of population. Theoretical and empirical researches have shown that EDAs are a class of efficient black-box optimization algorithms.

The core of EDAs is the probabilistic model.

For 0-1 domain problem, the basic probabilistic model is very simple:  $P(0) = p$  and  $P(1) = 1 - p$ , where  $0 \leq p \leq 1$ . All the existent 0-1 EDAs are based on the basic model. During the last decade, large amounts of different versions of 0-1 EDAs were developed, which can be classified into three categories [1]: without interaction, pairwise interaction and multi-interaction. PBIL, UMDA and cGA are estimation of distribu-

---

<sup>\*</sup> The work is funded by National Key Project for Basic Research of China (G2002cb312205).

tion algorithms which do not take the interaction among variables into account; BMDA and MIMIC use probabilistic model that can represent the relationship between two variables; BOA and FDA can describe the distribution of solutions by Bayesian Network, which can model complex interaction among variables [1-3]. The amazing success in discrete domain attracts people to design efficient EDAs for continuous problems.

However, choosing probabilistic model for continuous domain is still a challenging problem, even though several attempts have been made to extend the research results from discrete to continuous problems. The complexity of continuous fitness landscape makes it impossible to choose an almighty probabilistic model that fits any problem. In general, continuous EDAs can be classified into two approaches – indirect and direct. The former employs transform methods such as discretization [4] and the latter estimates the parameters of the predefined distribution [3,6-9]. It should be noted that the “direct” approach occupies a predominant position because the “indirect” approach fails to scale with problem size and solution precision [12]. According to the probabilistic model employed in “direct” approach, continuous EDAs can be further classified into two categories: Gaussian-based EDAs[1,3,6-9] and Histogram-based EDAs [5]. Most of the work concentrates on Gaussian probabilistic model. These include PBIL<sub>C</sub>, UMDA<sub>C</sub>, EMNA, EGNA, IDEA and so on [1,3]. Histogram-based EDAs mainly refer to the work by Tsutsui, S. et al.[5], in which marginal histogram model was used to model the population in continuous domain for the first time.

The purpose of the paper is to further study the continuous EDAs based on histogram model. A novel estimation of distribution algorithm (HEDA) based on histogram model is developed. Accumulation strategy is used to update the probabilistic model. In sampling phase, mutation strategy is designed to enhance the population diversity. Experimental analyses will show the performance of HEDA compared with FWH, SGA, UMDA-Gaussian[7] and CMA-ES[11].

The present paper is organized as follows. Next section will give the detailed description of HEDA. In Section 3, numerical experiments of HEDA, FWH, CMA-ES, UMDA-Gaussian and SGA are described. The paper is concluded in Section 4.

## 2 HEDA – Histogram-Based Estimation of Distribution Algorithm

### 2.1 The General Description of HEDA

- (1) Set the generation counter  $t := 1$
- (2) Divide the searching space of each variable into a certain number of bins. These bins should be of the same width and do not overlap with each other.
- (3) Initialize the histogram model in which the height of each bin is same. The histogram generated should be normalized.
- (4) Generate population  $P(t)$  using sampling method described in subsection 2.3.
- (5) Evaluate and rank the population  $P(t)$ . Save the elitist.
- (6) Update the histogram model using accumulation learning strategy.

- (7) Update the generation counter  $t := t + 1$ .
- (8) If the termination conditions have not been satisfied, go to step 4.
- (9) HEDA is finished and solutions are obtained in  $P(t)$ .

## 2.2 Learning Method in HEDA

Accumulation learning strategy is proposed to update the histogram model. The histogram model is updated according to two kinds of information: historical and current information. In each generation, for each variable  $i$ , the selected individuals will be used to construct a histogram model  $H_c^i$ . The old histogram for variable  $i$  is denoted as  $H_h^i$ . The height of a certain bin  $j$  of the renewed histogram for variable  $i$  is:

$$H^i(j) = \alpha H_h^i(j) + (1 - \alpha) \cdot H_c^i(j), \quad (1)$$

where  $H_h^i(j)$  is the height of bin  $j$  in the old model,  $\alpha$  ( $0 \leq \alpha \leq 1$ ) is the accumulation factor which determines the effect of the old model on the new model,  $H_c^i(j)$  is the height of the bin  $j$  in model  $H_c^i(j)$ , and  $\sum_j H_c^i(j) = 1$ . It is clear that new histogram  $H^i$  has been normalized. Accumulation strategy used in HEDA is a method to reserve the information of historical model. In comparison with FWH [5] which reserves some good individuals in each generation, HEDA emphasizes the importance of the model building. The reservation of the historical model also reserves the information of the good individuals in the past generations. If  $\alpha = 0$ , HEDA is reduced to the FWH [5].

In addition, the contribution of different individuals with different fitness values is considered in the learning process. The histogram model is learned according to relative ranking among different individuals. The height of each bin is modified according to the ranking and the position of each individual. Different rankings of the individuals lead to different increments of the bins. In the paper, the relationship is linear. If  $N$  best individuals of the population are selected, the  $k$ -th best individual ( $k \leq N$ ) will make an increment of the corresponding bin which it belongs to by:

$$\Delta h_k^i = (N - k + 1) / \sum_{l=1}^N l = \frac{2(N - k + 1)}{N(N + 1)} \quad (2)$$

So,

$$H_c^i(j) = \sum_{k=1}^N \Delta h_k^i \cdot \delta_{jk}^i \quad (3)$$

where  $\delta_{jk}^i = 1$  for  $\{\delta_{jk}^i \mid k \in \{1, 2, \dots, N\} \wedge \min_j^i \leq v_k^i < \max_j^i\}$ , and  $\delta_{jk}^i = 0$  otherwise.  $v_k^i$  denotes the value of variable  $i$  of the  $k$ -th best individual,  $\min_j^i$  and  $\max_j^i$  denote the lower and upper bound of bin  $j$  of variable  $i$ . The better individuals will have more effect on the new model. Updating  $H_c^i$  based on the ranking information helps improve the convergence property of HEDA.

### 2.3 Sampling Method in HEDA

In HEDA, the population is sampled according to the model as follows: First, the bin  $j$  is selected according to the probability of  $H^i(j)$ ; then an individual is generated in the searching space of the bin  $j$  with uniform distribution.

In order to enhance the diversity of population, mutation strategy is used. In HEDA, the mutation strategy means that each variable of an individual has a probability to be generated randomly. Here “randomly” means that the variable of an individual is generated with uniform distribution in the whole searching space. If the mutation rate is set  $p_m$ , that means there is a possibility of  $p_m$  for each variable of each individual to be generated randomly and a possibility of  $1 - p_m$  for it to be generated according to the histogram model.

## 3 Numerical Experiments

### 3.1 Experimental Settings

Several well-known continuous test functions, which include the 20-variable two-peak function, the 20-variable Rastrigin function, the 10-variable Griewank function and the 5-variable Schwefel function, are used to verify the performance of HEDA. Table 1 lists these experimental functions and their respective optimal solutions.

**Table 1.** Test functions

| Function name | Formulation  | Domain         | Optimal solution      |
|---------------|--|----------------|-----------------------|
| Schwefel      | $\sum_{i=2}^n ((x_1 - x_i^2)^2 - (x_i - 1)^2)$                                   | $[-2, 2]^5$    | $[1, 1, 1, \dots, 1]$ |
| Rastrigin     | $10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$                                 | $[-5, 5]^{20}$ | $[0, 0, 0, \dots, 0]$ |
| Griewank      | $\sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$ | $[-5, 5]^{10}$ | $[0, 0, 0, \dots, 0]$ |
| Two-peak      | $5n - \sum_{i=1}^n f_i$ , twopeak $f_i$ [5]                                      | $[0, 12]^{20}$ | $[1, 1, 1, \dots, 1]$ |

In the experiments, the proposed HEDA has been compared with several well-known continuous evolutionary algorithms including FWH, UMDA-Gaussian, SGA and CMA-ES. FWH is the original proposed continuous EDA based on histogram probability model [5]. The objective of comparison with FWH is to show that HEDA is a superior optimization method based on histogram model. UMDA-Gaussian is an EDA based on Gaussian probability model [7]; SGA refers to simple genetic algorithm; CMA-ES is an advanced evolution strategy with covariance matrix adaptation [11]. The objective of the comparison with UMDA-Gaussian, SGA and CMA-ES is to show that, among many kinds of continuous evolutionary algorithms, HEDA hold an outstanding position in solving the above test functions.

To evaluate the performance of HEDA, two criteria are used: convergence property and scalability. The convergence property is to measure the ability of the algorithm to reach the global optimum. In our experiments, we evaluate the convergence property by measuring the number of runs in which algorithm succeeds in finding the global optimum and the mean number of function evaluations (MNE) to find the global optimum in those successful runs. We define the successful detection of the solution as being within  $\pm\epsilon$  of the actual optimum point ( $\epsilon=0.1$ ). The scalability of HEDA in solving two-peak problem is used to see how the behavior of algorithms changes when the dimension of the problem increases.

All the algorithms run 20 independent times on each problem. The termination condition for the 5 algorithms is detection of optimal solution or maximum 100,000 function evaluations. In all the algorithms, the initial population is generated uniformly. The parameter settings HEDA, FWH, SGA, UMDA-Gaussian and CMA-ES are as follows.

HEDA: the width of bin is set 0.1, mutation rate is set 0.05, and accumulation rate 0.2. 50% of population is selected for model updating. Elitist strategy is used.

FWH: we directly use the results obtained by FWH published in [5].

SGA: crossover probability 0.8 and mutation probability 0.05. Selection method is tournament selection. Elitist strategy is used.

UMDA-Gaussian: the Gaussian probability model is updated using method introduced in [9]. 50% of parent population is selected for model updating. Elitist strategy is used.

CMA-ES:  $\mu = \lambda/4$ , where is  $\lambda$  is the size of parent population and  $\mu$  is the number of descendants. The Matlab code of CMA-ES in [13] is used in the experiments.

### 3.2 Convergence Property

Table 2 illustrates the convergence property of HEDA, UMDA-Gaussian, FWH, SGA, and CMA-ES in solving the above test functions.

Firstly, we compare the performance of HEDA and FWH, both of which have the common feature: estimation of distribution algorithms based on marginal histogram probability model. The results of FWH have been published in [1]. From table 2, it is obvious that HEDA performs much better than FWH. For example, in solving Griewank function, with population size 100, HEDA can always obtain the optimal solution in the 20 runs, while FWH fails in all the 20 runs. Excellent behavior in solving the 4 problems shows the effectiveness of accumulation learning strategy and mutation strategy in HEDA.

And then, let's see the comparison with SGA, UMDA-Gaussian and CMA-ES. As shown in Table 2, for Schwefel functions, CMA-ES exhibits very good performance. For example, with population size 100, CMA-ES can obtain optimal solution in all the 20 runs with less MNE 1205.6. UMDA-Gaussian performs very well in solving Griewank function and it can always succeed in converging to optimum in all the 20 runs. However, UMDA-Gaussian and CMA-ES performs worse than HEDA in solving other test functions. It can be observed that the convergence property of HEDA

**Table 2.** Convergence property of HEDA, FWH, SGA, UMDA-Gaussian and CMA-ES in solving a class of continuous optimization problems

|           |              | Population Size |         |      |         |     |         |      |         |     |         |      |         |
|-----------|--------------|-----------------|---------|------|---------|-----|---------|------|---------|-----|---------|------|---------|
|           |              | 100             |         |      | 200     |     |         | 300  |         |     | 400     |      |         |
|           |              | #OPT            | MNE     | #OPT | #OPT    | MNE | MNE     | #OPT | MNE     | MNE | #OPT    | #OPT | MNE     |
|           |              |                 |         |      |         |     |         |      |         |     |         |      |         |
| HEDA      | Schwefel     | 20              | 2607.5  | 20   | 3221.4  | 20  | 2475.3  | 20   | 4124.5  | 20  | 3985.6  | 20   | 5426.7  |
|           | Rastrigin    | 20              | 23756.0 | 20   | 19601.2 | 20  | 17638.2 | 20   | 16968.2 | 20  | 17888.4 | 20   | 16397.5 |
|           | Griewank     | 20              | 6164.3  | 20   | 5709.4  | 20  | 5687.7  | 20   | 6972.0  | 20  | 9218.7  | 20   | 10081.2 |
|           | Two-peak     | 20              | 15422.7 | 20   | 12442.2 | 20  | 11874.4 | 20   | 12060.0 | 20  | 10456.3 | 20   | 12112.3 |
| FWH       | Schwefel[5]  | 16              | 1342.6  | 19   | 1962.3  | 19  | 3302.9  | 20   | 4012.7  | 20  | 5924.8  | 20   | 7597.9  |
|           | Rastrigin[5] | 1               | 3478.0  | 13   | 6770.6  | 20  | 9720.2  | 20   | 12616.4 | 20  | 18518.5 | 20   | 24113.5 |
|           | Griewank[5]  | 2               | 2779.0  | 7    | 5783.3  | 15  | 9220.9  | 19   | 12439.8 | 19  | 17517.2 | 19   | 22502.3 |
|           | Two-peak[5]  | 1               | 2562.0  | 14   | 4926.5  | 20  | 7178.3  | 19   | 9609.8  | 19  | 14016.6 | 20   | 18537.9 |
| UMDA-Gaus | Schwefel     | 18              | 1460.0  | 6    | 3666.7  | 9   | 4560.0  | 7    | 4666.8  | 4   | 7240.8  | 5    | 8056.0  |
|           | Rastrigin    | 0               | ----    | 6    | 18600.0 | 12  | 28450.4 | 11   | 36320.4 | 13  | 57632.0 | 14   | 68572.4 |
|           | Griewank     | 20              | 1840.5  | 20   | 3690.4  | 20  | 5490.6  | 20   | 7250.4  | 20  | 10635.0 | 20   | 13349.8 |
|           | Two-peak     | 0               | ----    | 0    | ----    | 0   | ----    | 0    | ----    | 1   | 9948.0  | 4    | 9680.0  |
| SGA       | Schwefel     | 4               | 2375.6  | 14   | 2957.2  | 17  | 4923.6  | 15   | 6826.8  | 20  | 9090.0  | 20   | 10142.5 |
|           | Rastrigin    | 0               | ----    | 0    | ----    | 0   | ----    | 5    | 27834.6 | 6   | 39212.8 | 6    | 50467.3 |
|           | Griewank     | 0               | ----    | 1    | 7564.0  | 2   | 14581.2 | 3    | 15067.8 | 5   | 30075.0 | 6    | 44612.5 |
|           | Two-peak     | 0               | ----    | 0    | ----    | 0   | ----    | 0    | ----    | 4   | 54326.0 | 7    | 93672.4 |
| CMA-ES    | Schwefel     | 20              | 1205.6  | 20   | 2035.4  | 20  | 2728.3  | 20   | 3181.5  | 20  | 4038.5  | 20   | 5352.2  |
|           | Rastrigin    | 0               | ----    | 1    | 19200.0 | 1   | 28600   | 0    | ----    | 6   | 55811.5 | 7    | 75220.3 |
|           | Griewank     | 12              | 3860.0  | 20   | 7429.5  | 20  | 10740.6 | 20   | 13525.0 | 20  | 18465.2 | 20   | 22450.6 |
|           | Two-peak     | 0               | ----    | 0    | ----    | 0   | ----    | 0    | ----    | 0   | ----    | 0    | ----    |

Note: “----” indicates that the algorithm failed to obtain the optimum in all the 20 runs.

is consistently good in solving all the 4 functions. CMA-ES often fails in solving Two-peak function and Rastrigin function; UMDA-Gaussian often fails in solving Two-peak function, Schwefel function and Rastrigin function. Compared with SGA, HEDA performs much better both in MNE and #OPT. Experimental results show that HEDA is a stable, robust and efficient algorithm in solving the test functions.

3.3 Scalability of HEDA

The scalability of HEDA is tested on Two-peak function. The problem dimension starts at 10, increased to 100 with step 10. For each dimension, 20 independent runs are executed. Fig.1 shows the mean number of fitness evaluations until HEDA finds the optimal solution. The number of fitness evaluations can be approximated by  $O(n^{1.256})$ . Therefore, the results indicate that HEDA can solve Two-peak function in sub-quadratic number of evaluations.

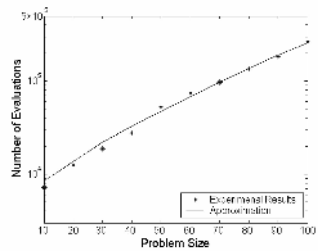


Fig. 1. Scalability of HEDA for Two-peak problem

3.4 Drawback of HEDA

In the previous subsections, experiments demonstrate that HEDA can efficiently solve the test functions and performs much better than some of well-known continuous evolutionary algorithms. However, in the procedure of HEDA, the relationship among variables is not taken into consideration. This will limit the ability of HEDA in solving complicated problems with strong linkage information. Here, 20-variable Rosenbrock function with optimum [1,1,...,1] is used to test the limited performance of HEDA:

$$f(x) = \sum_{i=2}^n 100(x_i - x_{i-1}^2)^2 + (1 - x_{i-1})^2 .$$

There are strongly correlated variables in Rosenbrock function. In the experiment, 10 runs are executed for each population setting. The experimental results in Table 3 demonstrate it is hard to find optimal solution using HEDA. The drawback of HEDA is due to the fact that HEDA does not encode the linkage information.

Table 3. Convergence property of HEDA in solving Rosenbrock function

| 100  |         | 200  |         | 400  |      | 800  |         |
|------|---------|------|---------|------|------|------|---------|
| OPT# | MNE     | OPT# | MNE     | OPT# | MNE  | OPT# | MNE     |
| 1    | 29232.6 | 2    | 18625.4 | 0    | ---- | 1    | 62816.0 |

## 4 Conclusion

In this paper, we have proposed an improved estimation of distribution algorithm (HEDA) based on histogram probabilistic model. In the algorithm, a novel learning strategy named accumulation strategy is proposed, which considers the historical and current information of population at the same time. Mutation strategy is brought into the sampling phase to enhance the population diversity. HEDA is tested to solve a class of well-known continuous problems. Experimental results demonstrate that HEDA significantly outperforms FWH, and exhibits excellent capability compared to other evolutionary algorithms, e.g. CMA-ES, SGA and UMDA-Gaussian.

It is also noted that the capability of HEDA is limited by the fact that HEDA does not take the relationship among variables into account. Future work will focus on extensions of HEDA that can model the relationship among variables.

## References

1. Larrañaga, P., Lozano, J. A.: Estimation of Distribution Algorithms: A new Tool for Evolutionary Computation. Kluwer Academic Publishers (2002)
2. Pelikan, M.: Hierarchical Bayesian optimization algorithm: Toward a new generation of evolutionary algorithms. Springer-Verlag (2005)
3. Očenášek, J.: Parallel Estimation of Distribution Algorithms. Ph.D. Dissertation in Brno University of Technology (2002)
4. Pelikan, M., David E.G., Tsutsui, S.: Combining the Strengths of the Bayesian Optimization Algorithm and Adaptive Evolution Strategies. IlliGAL Report No. 2001023 (2001)
5. Tsutsui, S., Pelikan, M., Goldberg, D.E.: Evolutionary Algorithm Using Marginal Histogram Models in Continuous Domain. IlliGAL Report No. 2001019, UIUC (2001)
6. Larrañaga, P., Etxeberria, R., Lozano, J. A., Peña, J. M.: Optimization by Learning and Simulation of Bayesian and Gaussian Networks. Technical Report EHU-KZAA-IK-4/99, University of the Basque Country (1999)
7. 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. Las Vegas, Nevada (2000)
8. Sebag, M., Ducolombier, A.: Extending population-based incremental learning to continuous search spaces. Parallel Problem Solving from Nature – PPSN V, Springer-Verlag (1998) 418–427
9. Larrañaga, P., Lozano, J. A., Bengoetxea, E.: Estimation of Distribution Algorithms based on multivariate normal and Gaussian networks. Technical report KZZA-IK-1-01, University of the Basque Country (2001)
10. Bosman, P., Thierens, D.: Expanding from Discrete to Continuous Estimation of Distribution Algorithms: IDEA. Parallel Problem Solving From Nature- PPSN VI (2000) 767–776
11. Hansen, N., Mueller S.D., Koumoutsakos, P.: Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). Evolutionary Computation, Vol. 11. (2003) 1–18
12. Ahn, C.W.: Real-coded Bayesian Optimization Algorithm. Advances in Evolutionary Algorithms: Theory, Design and Practice. Springer Publication, Vol.18. (2006) 85–124
13. URL: <http://lautaro.bionik.tu-berlin.de/user/niko/>