Proceedings of the 7th Asian Control Conference,
Hong Kong, China, August 27-29, 2009

Evolutionary Algorithm using Kernel Density Estimation Model in Continuous Domain

Na Luo and Feng Qian

Abstract—**Estimation of Distribution Algorithm (EDA) is a kind of evolutionary algorithm which updates and samples from probabilistic model in evolutionary course. The key of EDA is the construction of probability model suitable for real distribution. Gaussian distribution is widely used in EDAs but the assumption of normality is not realistic for many real-life problems. In this paper, a new EDA using kernel density estimation (KEDA) is introduced. Adaptive change strategy of kernel width is presented and selection scheme, sampling method are also given cooperated with KEDA. The results of 5 benchmark functions show that results of KEDA outperform $\mathbf{PBIL_C}$, $\mathbf{UMDA_C}$, $\mathbf{EDA^G}$, H-EDA.**

I. Introduction

Estimation of Distribution Algorithm (EDA) is a kind of evolutionary algorithm which updates probabilistic model of individuals in each generation, and then make new samples according to the probabilistic model. EDA has strong theoretical basis. It is proposed that $x = (x_1, x_2, \cdots, x_n)$ is the vector of problem and $p^s(x,k)$ is the real distribution of the current population. It has been proved that if distribution of next generation $p(x,k+1) = p^s(x,k)$ and the selection method is proportion, truncation or tournament, EDA can converge to global optimization [1]. In essence, EDA needs to estimate the density function $p(x,k+1)$ close to $p^s(x,k)$ from underlying population with receptible computation cost.

The problem of density estimation has been well studied in statistics. There are basically two approaches. (i) parametric density estimation and (ii) non-parametric density estimation. The first approach assumes that the form of distribution is known, and the problem is simplified by learning the parameters from one of the known density functions. The second approach does not make any assumption about the structure of the function and in general, it has higher computational requirements.

In continuous EDAs, Gaussian distribution is popular and widely used. It is used as a parametric distribution method. However, the assumption of normality is not realistic for many real-life problems. Hybrid gaussian model is proposed to compensate shortage of gaussian model [2]. As non-parametric density estimation, histogram model has been used in EDA without any assumption of population distribution [3,4]. But with increment of problem size, histogram scale up exponentially. Comparing with histogram model, Kernel density estimation is of more interesting properties. In order to efficiently obtain global optimal solutions, kernel density estimation is used with estimation of distribution algorithm (KEDA) in this paper.

The reminder of this paper is organized as follows. Section II is a short review of standard EDA and kernel density estimation method. Section III gives a description of a new algorithm named kernel density estimation of distribution algorithm. In Section IV, we give experiment and the analysis of the new algorithm. And we give our conclusion in Section V.

II. Retrospect of KEDA

A. Estimation of Distribution Algorithm in Continuous Domain

Estimation of Distribution Algorithms were proposed originally for combinatory optimizations. Now they have been studied intensively and extended to continuous optimization. Many approaches have been proposed in the framework of EDAs. They are similar in the way that a probability distribution is estimated from the selected individuals. According to the complexity of variable dependency, they can be categorized into three groups: (i) univariate dependency, such as PBIL [5], CGA [6], UMDA [7], (ii) bivariate dependencies, such as MIMIC [8], COMIT [9], BMDA [10], and (iii) multivariate dependencies, such as ECGA [11], BOA [12], FDA [13].

Manuscript received January 31, 2009. This work was funded by Distinguished Young Scholars Program of the National Natural Science Foundation of China (NSFC) (Grant Number: 60625302), National Natural Science Foundation of China (General Program, No:60704028), Major State Basic Research Development Program of Shanghai(07JC14016), PCSIRT, Supported by the 111 Project(B08021), Supported by Shanghai Leading Academic Discipline Project(Project number: B504), National High-Tech Research and Development Program of China (863 Program, No: 2006AA04Z168,2007AA04Z193).

Na Luo is with Automatic Institute, Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, State-Key Laboratory of Chemical Engineering, East China University of Science and Technology, Shanghai, 200237 China (e-mail: naluo@ ecust.edu.cn).

Feng Qian is with Automatic Institute, Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, State-Key Laboratory of Chemical Engineering, East China University of Science and Technology, Shanghai, 200237 China . (phone: 86-21-64252060; fax: 86-21-64252056; e-mail: fqian@ ecust.edu.cn)..

978-89-956056-9-1/09/©2009 ACA

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.

Larrañage et al. extended UMDA to Continuous domain [14] ($UMDA_C$). Gaussian network is constructed as a probabilistic graphic model and used in EDAs. $PBIL_C$ [15], $MIMIC_C$ [14], EGNAee [14] are all extension to continuous domain. All of these algorithms are based on gaussian network. In the next, $UMDA_C$ is illustrated as an example.

$UMDA_C$ models the univariate function with the Gaussian distribution and uses the maximum likelihood to estimate the parameters. It is assumed that the distribution is Gaussian distribution. So $UMDA_C$ has the following form of the univariate density function:

$$P(x_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{1}{2\sigma_i^2}(x_i-\mu_i)^2} \tag{1}$$

Where μ_i and σ_i refer to the mean and the variance respectively. It is denoted as $P(x_i) \sim N(\mu_i,\sigma_i^2)$. For samples, maximum likelihood is used to estimate parameter of mean value μ_i and variance σ_i^2。

$$\hat{\mu}_i = \frac{1}{n}\sum_{k=1}^{n} x_{ik} \tag{2}$$

$$\hat{\sigma}_i^2 = \frac{1}{n}\sum_{k=1}^{n} (x_{ik} - \hat{\mu}_i)^2 \tag{3}$$

Where x_{ik} is sample of variant x_i, k is the number of samples, $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ are estimation value of μ_i and σ_i^2.

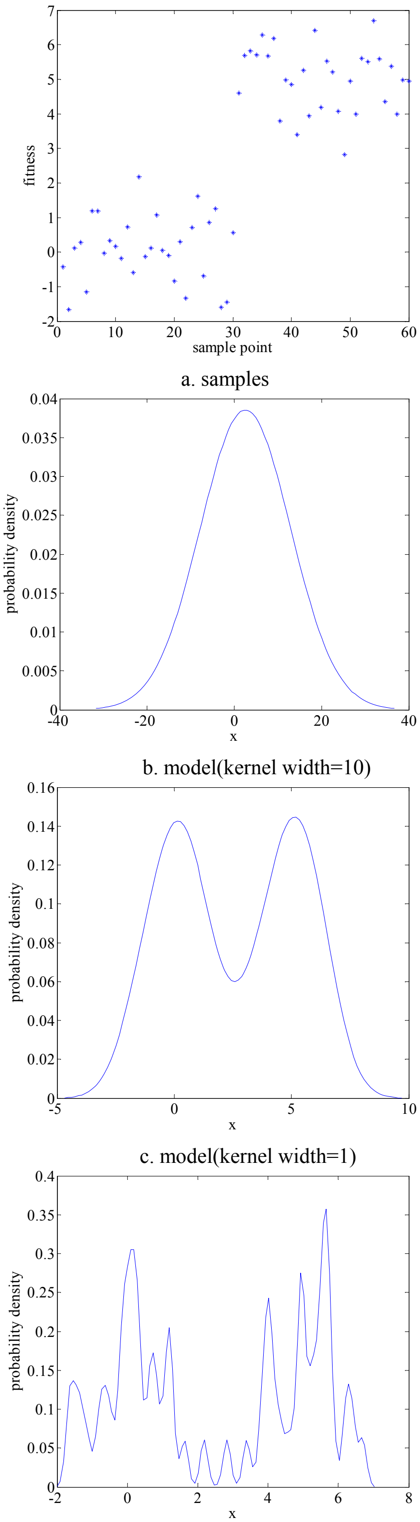
It is not so difficult to sample from a normal distribution. The framework of $UMDA_C$ is as follows:

1) Initialize a population of individuals randomly.
2) Select individuals according to a selection method.
3) Estimate n-dimensional Probability Density Function using gaussian model.
4) Sample new individuals from gaussian model as new population.
5) New population is partially or not replacing by the elitism individuals.
6) Stop if some stopping criterion is reached, go to step 2 otherwise.

As many evolutionary algorithms, EDAs using gaussian model in continuous domain has some disadvantages, such as premature. Many improvements are proposed in how to set variance. Making variance scaling adaptive [16], repairing covariance matrix [17] could get some effect in solving the problem. But in essence, Gaussian distribution assumption is not suitable for most real problems. Next a non-parametric probability estimation method name kernel density estimation is introduced.

B. Kernel Density Estimation

It is a smooth continuous function using kernel density estimate of samples distribution. Let $x_1, x_2, \cdots, x_n$ are independent samples from the same distribution $f(x), x \in R$. If there exists a bounded function $K(u) \geq 0$ which satisfied following conditions:



a. samples

b. model(kernel width=10)

c. model(kernel width=1)

d. model(kernel width=0.1)

Fig. 1. Distribution model of samples

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.

(1) $\int_{-\infty}^{+\infty}|k(u)|du < +\infty$ *(2)* $\lim_{|u|\to\infty} uk(u) = 0$ *(3)* $k(-u) = k(u)$ *(4)* $\int_{-\infty}^{+\infty}k(u)du = 1$

The estimation of distribution f(x) is given as:

$$\hat{f}(x) = \frac{1}{nh_n}\sum_{i=1}^{n} K(\frac{x-x_i}{h_n}) \quad (4)$$

Where $K(u)$ is a kernel function, h_n is kernel width which is a smooth parameter which is related with n. Kernel functions can be selected as uniform, triangle, epanechikov, quadratic, tri-weight, gauss, cosinus, and exponent. The most commonly used kernel function is gaussian kernel:

$$k(x, x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-x_i)^2} \quad (5)$$

For each kernel function can guarantee stable property of the density estimation, the selection of kernel function is not the key factor in kernel density estimation. What is important is the selection of kernel width. In figure 1, it is illustrated of different distribution model for the same samples.

It is clear in figure 1 that the kernel width intensively influences distribution function. If the amount of samples is large, the kernel width should be large. If the kernel width is too large, the distribution density is average and the resolution of the model is low. If the kernel width is too little, the influence of noise is too large and statistic change of distribution model is large.

As to the selection of kernel width, there are 3 kinds of optimal criterion: *(1)* minimize mean square error, *(2)* minimize integral of mean square error and *(3)* minimize gradual integral of mean square error.

III. Kernel-density Estimation of Distribution Algorithm

The core concept of KEDA is to use kernel density estimation as probability model method in the optimization algorithm. During the course of model selection samples, kernel density estimation has implicit property which can cluster samples automatically. Figure 2 shows the change of new generation in optimization of Griewank multimodal function.

Because of adaptive clustering in EDA, the algorithm can find many local optimal values and it is benefit for search of global optimal. The framework of KEDA is as follows:

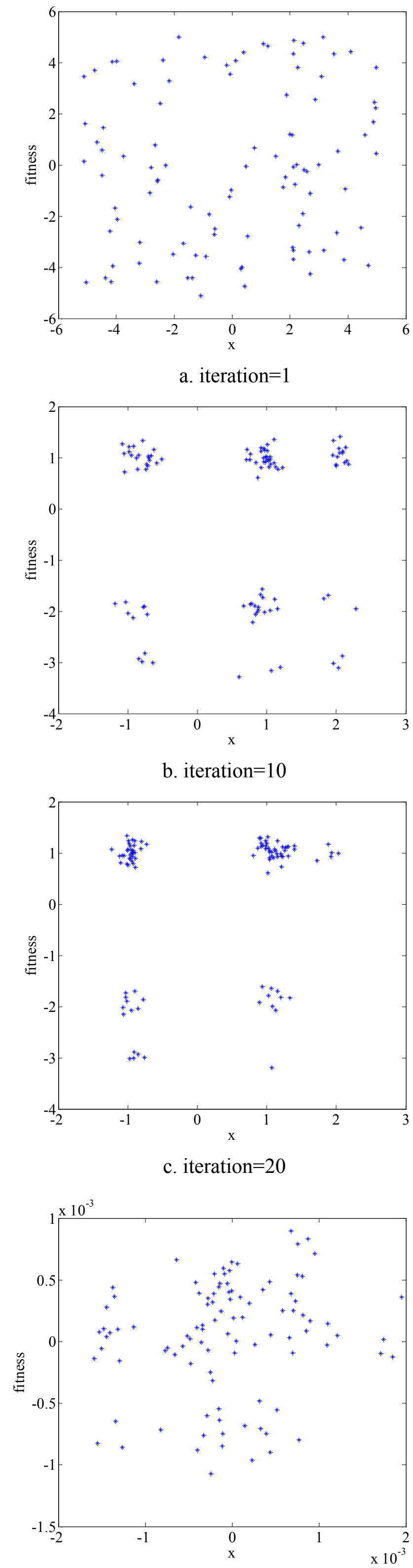
(1) Initialize a population of N individuals randomly.

(2) Calculate the fitness of population and select $S \le N$ individuals (D_S) according to some selection scheme.

(3) Select e individuals (D_e) as elitisms according to fitness.

(4) Compute kernel width according to D_S.

(5) Estimate kernel density model $\hat{p}(x) = \frac{1}{nh_n}\sum_{i=1}^{n} K(\frac{x-x_i}{h_n})$ from D_S.



a. iteration=1

b. iteration=10

c. iteration=20

d. iteration=1000

Fig. 2. Distribution of new individuals using kernel density estimation

(6) Sampling N individuals from $\hat{p}(x)$ using stochastic universal sampling.

(7) Replace e worst individuals using D_e *(8)* Stop if some stopping criterion is reached; otherwise go to *(2)*.

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.

A. selection of kernel width

In this paper, gaussian kernel is used as kernel function. Considering evolving course, kernel width is decided dynamically during the optimization. There are two methods to fix kernel width:

1) Change with Iteration

It is clear that the search domain becomes smaller while iteration incremented. So it is a good solution that kernel width changes with iteration linearly or nonlinearly, illustrated as in equation (6) and (7). The rule can be used aperiodically.

$$w(t)=\frac{(w_i-w_e)(T_{\max}-t)}{T_{\max}}+w_e \tag{6}$$

$$w(t)=\left(\frac{t-1}{T_{\max}-1}\right)^{\lambda}\left(w_i-w_e\right)+w_i \tag{7}$$

Where w_i is denoted as initial kernel width, w_e is denoted as the end width. $T_{\max}$ is the maximum iteration and t is current iteration.

2) Adaptive Kernel Width Change

Kernel width can be obtained using feedback information of new generation. In this paper, the selection individuals are samples. Average distance of samples and its k neighbors can be calculated as kernel width, as follows:

$$w(t)=\frac{1}{n}\sum_{j=1}^{n}\frac{1}{k}\sum_{i=j-k/2}^{i=j+k/2}x_i \tag{8}$$

B. Sampling method

Sampling method is a key in KEDA. Reducing sampling error is the standards of choosing sampling method. We used extended stochastic universal (E-SUS) in [3] as our sampling method. E-SUS comes from Baker's stochastic universal sampling which was proposed for the proportional selection operator. E-SUS extend it. When used in this paper, the kernel density function is necessary to be divided into many bins. For each new individual, each value is generated as follows: first a bin is selected according to the probability of each bin. Then the value of variable is generated by generating a number from the selected bin with uniform distribution. This is repeated until all individuals are obtained. The simplest method to sample each bin is to use a roulette wheel. The matlab pseudo code is in figure 3.

C. Selection scheme

From the generated population, selection is performed to shortlist promising individuals S. Many methods exist to select individuals, such as tournament selection or roulette wheel selection. It is important to determine the size S. if it is too small, it may not include all promising search points. If it is too large, it increase the learning overhead. KEDA select S with fixed size using truncate selection.

```
% S: sampling individual number, n: number of variables, H: number of bins
% p[j][h]: probability density of bin h of variable xj
% l[j][h]:  left edge position of of bin h of variable xj
% v[S][n]: array for sampled vectors
for j=1:n
  ptr=rand(1,1);
  sum1=0;
  k=1;
  xh=randperm(S);
  for h=1:H
      expected=p(j,h)*S; %
      sum1=sum1+expected;
      while sum1>ptr
          ptr=ptr+1;
          v(xh(k),j)=l(j,h)+(l(j,h+1)-l(j,h))*rand(1,1);
          k=k+1;
      end
  end
end
```

Fig. 3. Pseudo matlab code of the E-SUS

D. Replacement

In the kernel estimation of distribution algorithm, precede population will be in part replaced to form the new population. Replacement operation can be represented similar to the selection operator. In KEDA, the elitism individual is selected from the precede populations proportional to fitness value.

IV. Experimental Analysis

In this paper 5 benchmark functions are selected to test KEDA comparing with $UMDA_C$, $PBIL_C$, EDA^G and H-EDA (Histogram Estimation of distribution algorithm). All these functions are tested for 50 times to obtain the max, min, mean and standard deviation of test results. The population of the algorithms is set 100, function dimension 30, iteration 4000. In all these 5 benchmark functions, there are 2 single modal functions and 3 multi-modal functions. They are illustrated in Table I.

In Table I, function f_1 and f_2 are unimodal problems. Quadric is a transformation of Sphere modal $f=\sum_{i=1}^{n}x_i^2$ with interdependence of variables. Rosenbrock function is a classical complex optimization problem. Its global optimal point is in a smooth, long and narrow curve. For there is little heuristic information, it is difficult to search the global optimal. Function f_3, f_4 and f_5 are classical nonlinear multimodal problems and they all have broad search space, many local optimal points and obstacles. The experimental results are summarized in Table II.

It can be seen from Table II that KEDA outperform in both unimodal and multimodal problems. Using appropriate kernel width, KEDA can model probabilistic model close to real distribution. Although Quadric function have only one

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.

TABLE I. BENCHMARK FUNCTIONS USED IN THIS STUDY

f1~f2 ARE UNIMODAL FUNCTIONS AND *f3~f5* ARE MULTIMODAL FUNCTIONS

Function	Name	Dim.	Search Space	Min/Best Position
$f_1 = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2$	Quadric	30	$(-100,100)^N$	0 (0,…,0)
$f_2(x) = \sum_{i=1}^{n-1}\left(100\left(x_{i+1}-x_i^2\right)^2 + \left(1-x_i\right)^2\right)$	Rosenbrock	30	$(-50,50)^N$	0 (1,…,1)
$f_3 = \frac{1}{4000}\sum_{i=1}^{n}(x_i)^2 - \prod_{i=1}^{n}\cos\left((x_i)/\sqrt{i}\right)+1$	Griewank	30	$(-300,300)^N$	0 (0,…,0)
$f_4 = \sum_{i=1}^{n}\left(x_i^2 - A\cos(2\pi x_i) + A\right), A=10$	Rastrigrin	30	$(-5.12,5.12)^N$	0 (0,…,0)
$f_5 = \sum_{i=1}^{n-1}\left(x_i^2+x_{i+1}^2\right)^{0.25}\times\left[\sin\left(50\times\left(x_i^2+x_{i+1}^2\right)^{0.1}\right)+1.0\right]$	Schaffer's f7	30	$(-100,100)^N$	0 (0,…,0)

TABLE II. PERFORMANCE COMPARISON BETWEEN ALGORITHMS

Function	Algorithm	Dim	Population	Iter	Mean	Std	Min	Max
Quadric	KEDA	30	300	2000	0.0001	0.0000	0.0001	0.0003
	$PBIL_C$	30	300	2000	1.6989	0.3199	1.0973	2.4978
	$UMDA_C$	30	300	2000	1097.7179	385.1003	451.3361	2069.5644
	EDA^G	30	300	2000	159.6986	80.6321	33.0753	410.3452
	H-EDA	30	300	2000	1875.8651	709.8346	610.1958	3811.6375
Rosenbrock	KEDA	30	300	2000	21.6218	30.3355	0.0010	109.5761
	$PBIL_C$	30	300	2000	377.0089	627.5467	84.9266	3371.9126
	$UMDA_C$	30	300	2000	32.6901	16.6895	27.2021	102.6613
	EDA^G	30	300	2000	29.2310	7.2152	27.4891	73.8248
	H-EDA	30	300	2000	7904.1345	11118.7186	240.9728	57204.5440
Griewank	KEDA	30	300	2000	0.0002	0.0000	0.0002	0.0003
	$PBIL_C$	30	300	2000	0.0511	0.0130	0.0252	0.0896
	$UMDA_C$	30	300	2000	0.0000	0.0000	0.0000	0.0000
	EDA^G	30	300	2000	0.0000	0.0000	0.0000	0.0000
	H-EDA	30	300	2000	0.7971	0.2830	0.1911	1.2372
Rastrigrin	KEDA	30	300	2000	0.0002	0.0000	0.0002	0.0003
	$PBIL_C$	30	300	2000	240.7583	32.2661	173.4078	303.0853
	$UMDA_C$	30	300	2000	2.7295	1.4059	0.0000	5.9698
	EDA^G	30	300	2000	1.8707	1.4859	0.0000	4.9748
	H-EDA	30	300	2000	77.7356	9.0495	57.4869	99.8036
Schaffer's f7	KEDA	30	300	2000	0.2795	0.9347	0.0009	4.9061
	$PBIL_C$	30	300	2000	130.6817	42.8877	4.2960	170.7498
	$UMDA_C$	30	300	2000	0.0477	0.0417	0.0008	0.1580
	EDA^G	30	300	2000	0.0182	0.0226	0.0002	0.0972
	H-EDA	30	300	2000	3.5644	2.0354	0.9121	12.3649

optimal, interdependence added make it difficult to find global optimum. From Table 3, it appears that KEDA can outperform to other EDAs on Quadric function. It can be noticed that among the EDAs, the algorithms with single Gaussian distribution assumption perform better than H-EDA because the bin width is not so appropriate for the problem with fixed value. But for KEDA, adaptive kernel width strategy gives good performance on this function. For Rosenbrock function, these EDAs perform badly because all these EDAs ignore interdependency information. Comparing to others, KEDA appeared not so bad although it can't find the best solution yet. This perhaps should be attributed to the sampling method E-SUS. During the sampling, it randomly matched variables. This also told us that distribution of interdependence should be used in KEDA in the future. For multimodal functions as Griewank, Rastrigrin, Schaffer's f7, KEDA can close to the global optimal and the standard deviation is small. This means that the stability of KDEA is good and it can jump out of the local optimal point with more chance. The trails of $PBIL_C$, $UMDA_C$, EDA^G, H-EDA,KEDA to 30 dimensions Quadric function is illustrated in figure 4.

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.

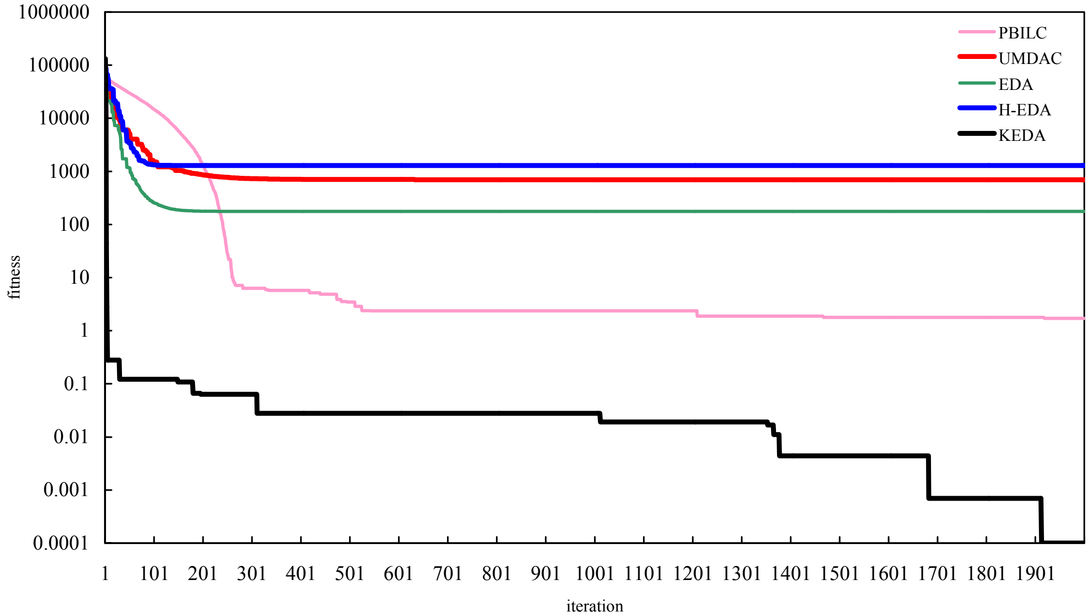


Fig.4. Optimization trail of $PBIL_C$, $UMDA_C$, EDA^G, H-EDA, KEDA to 30 dimensions Quadric function

V. Conclusion

Up to now most research works use Gaussian distribution as model to construct new algorithms in EDA category. For real problem, normal gaussian assumption is not always suitable. This paper gives kernel density estimation as model construction method and test of unimodal and multimodal test functions have been shown good performance of this method.

For it does not consider interdependence between variables, multivariate kernel density estimation will be used in the future work of EDAs.

References

[1] Zhang Q.F. and Heinz Mühlenbein, "On convergence of a class of optimization algorithms using estimation of distribution", *IEEE Transactions on Evolutionary Computation*, vol. 8,no. 2,pp.127-136,2004.

[2] Bin Li, Run-tian Zhong, Xian-ji Wang, Zhen-quan Zhuang, "Continuous Optimization based-on Boosting Gaussian Mixture Model", in *Proceeding of the 18th International Conference on Pattern Recognition (ICPR 2006)*, Hong Kong, China,vol.1,2006, pp.1192 – 1195.

[3] Shigeyoshi Tsutsui, Martin Pelikan, and David E. Goldberg, "Evolutionary Algorithm using Marginal Histogram Models in Continuous Domain", In *Proc. of the 2001 Genetic and Evolutionary Computation Conference Workshop Program*,2001.

[4] Ding N，Zhou S D，Sun Z Q, "Histogram-Based Estimation of Distribution Algorithm: A Competent Method for Continuous Optimization Histogram-based estimation of distribution algorithm：a competent method for continuous optimization", *Journal of computer science and technology*, vol.23,no.1,pp.35-43,2008.

[5] Baluja S., "Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and competitive Learning", Technical Report CMU-CS-94-163, Pittsburgh, PA: Carnegie Mellon University, 1994.

[6] G.R. Harik, F.G. Lobo, and D.E.Goldberg, "The compact genetic algorithm", *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, vol. 3, no. 4, Nov,1999,pp.287-297.

[7] H.Mühlenbein. and Paaß G., "From recombination of genes to the estimation of distributions I. Binary parameters. *Lecture Notes in Computer Science*, vol.1141,Berlin,pp.178-187,1996.

[8] Jeremy S. De Bonet, Charles L. Isbell, Paul Viola, "MIMIC: finding optima by estimating probability densities," *Advances in Neural Information Processing Systems*, C. Mozer, Ed. Cambridge, MA: MIT Press, 1997, vol. 9, pp. 424-431.

[9] Baluja, S., Davies, S., "Combining multiple optimization with optimal dependency trees", Technical Report CMU-CS-97-157, Carnegie Mellon University, Pittsburgh,1997.

[10] M.Pelikan and H.Mühlenbein, "The bivariate marginal distribution algorithm", *Adv. Soft Comput. Eng. Design and Manuf.*, pp.521-535,1999.

[11] G. Harik. Linkage Learning via Probabilistic Modeling in the ECGA. Illegal Report No. 99010, University of Illinois at Urbana-Champaign, Urbana, IL, January 1999.

[12] M.Pelikan, D.E.Goldberg and E. Cantú-Paz, "BOA: The Bayesian optimization algorithm", In *Proc. Genetic and Evolutionary Computation Conf.(GECCO-99)*,vol. I,pp.525-532,1999.

[13] H.Mühlenbein. and Mahnig T., "Convergence theory and applications of the factorized distribution algorithm", *Journal of Computing and Information Technology*, vol.7,no.1,pp.19-32,1999.

[14] P. Larrañaga, R. Etxeberria, J.A. Lozano, J.M. Peña, "Optimization in continuous domain by learning and simulation of Gaussian networks", in *The Proceeding of the 2000 Genetic and Evolutionary Computation Conference Workshop Program*, Las Vegas, Nevada,2000,pp. 201–204.

[15] Michèle Sebag, Antoine Ducoulombier, "Extending Population-Based Incremental Learning to Continuous Search Spaces", *Lecture Notes In Computer Science*,vol.1498,pp.418-427,1998.

[16] Cai Yunpeng, Sun Xiaomin, Xu Hua, Jia Peifa, "Cross entropy and adaptive variance scaling in continuous EDA", in *GECCO'02*,2007,London, England, United Kingdom,pp.609-616.

[17] Weishan Dong, Xin Yao, "Covariance matrix repairing in Gaussian based EDAs", in *2007 IEEE Congress on Evolutionary Computation (CEC 2007)*,pp.415-422.

Authorized licensed use limited to: Universidad Pais Vasco. Downloaded on November 18,2025 at 22:56:59 UTC from IEEE Xplore. Restrictions apply.