A Restart Univariate Estimation of Distribution Algorithm: Sampling under Mixed Gaussian and Lévy probability Distribution

Yu Wang, *Student Member, IEEE*, Bin Li, *Member, IEEE*

Abstract— A univariate EDA denoted as "LSEDA-gl" for *large scale global optimization* (LSGO) problems is proposed in this paper. Three efficient strategies: sampling under mixed Gaussian and Lévy probability distribution, Standard Deviation Control strategy and restart strategy are adopted to improve the performance of classical univariate EDA on LSGO problems. The motivation of such work is to extend EDAs to LSGO domain reasonably. Comparison among LSEDA-gl, EDA with standard deviation control strategy only (EDA-STDC) and similar EDA version "continuous univariate marginal distribution algorithm" UMDAc is carried out on classical test functions. Based on the general comparison standard, the strengths and weaknesses of the algorithms are discussed. Besides, LSEDA-gl is tested on 7 functions with 100, 500, 1000 dimensions provided in the CEC'2008 Special Session on LSGO. This work is also expected to provide a comparison result for the CEC'2008 special session.

I. Introduction

The notion of modelling the search space was first proposed as an approach in the statistics and machine learning domain. Recently, many works within Evolutionary Computation community have employed probabilistic models to describe the solution space [12]. These methods have become known as Estimation of Distribution Algorithms (EDAs). EDAs have been applied with significant success to many numerical optimization problems with less than 100 dimensions in the past few years. Optimization by EDA can be summarized into two major steps:

- 1 Model the promising area of solution space of the optimization problem by learning from the superior solutions found thus far;
- 2 Generate the population (i.e., offspring) for the next generation by sampling under the estimated probabilistic model and then, replace the old population (i.e., parents) partially or fully.

These two steps can be regarded as a population-based version of the classical generate-and-test method [3]. As is shown in the above steps, there are no crossover or mutation operators in EDAs in contrast to other classical EAs. The evolution dynamics of EDAs depend on the distribution of the population directly. Therefore, the major advantage of EDAs is that they can explicitly learn the promising area of solution space of the optimization problem and then use this structural information to efficiently generate new individuals [14].

In the EDA domain, modelling the structure of the optimization problem accurately has been a highly concerned area recently. To overcome the disadvantage of limited learning ability by univariate EDAs, such as PBIL [15], SHCLVND [16] and UMDAc [17], EDAs whose dependencies are considered in terms of pairwire or multiwire when building probability model are proposed recently. The famous examples are MIMIC [17], EGNAee [17], CEGDA[4] and CEGNA [4]. Among the above algorithms, the learning ability of the EDAs is always considered as the major indicator of the performance. However, the fundamental task of EDAs is to search the global optimum of the given optimization problem, rather than to only model the structure of the optimization accurately. Further more, the computational cost of conducting a complex model is too expensive for LSGO problems, whose dimensional size (D) is usually more than 100. The reported studies on extending EDA to LSGO domain are scarce so far.

In this paper, a robust univariate EDA denoted as "EDA for large scalar global optimization" (LSEDA-gl) is proposed for LSGO problems. In LSEDA-gl, an effective strategy, sampling under mixed Gaussian and Lévy probability Distribution (SGLD), is introduced to balance the relationship between optimization and learning. Compared with the previous version of LSEDA "EDA-STDC" without SGLD [18], the convergence speed is accelerated by SGLD. It has been observed that one fatal problem of EDAs when it is applied to high dimensional problems is that standard deviations of some variables often shrink dramatically to zero while others are still large, this will cause the lost of diversity. For this reason, Standard Deviation Control strategy (STDC) is designed to improve the performance for univariate EDA on LSGO problems. In order to prevent the algorithm from converging to local optima, a restart strategy is also adopted here. The major idea behind these three strategies is to force the algorithms to sample individuals in a reasonably wider area, and therefore, to maintain the exploration ability effectively. Experimental results show that these three strategies can help the univariate EDA to solve the high dimensional unimodal and many multimodal problems efficiently and effectively.

The remainder of this paper is organized as follows: In section II, the principle of LSEDA-gl is introduced, including three strategies SGLD, STDC and restart strategy. In section

3917

III, comparison among LSEDA-gl, EDA-STDC and UMDAc is carried out to evaluate how the incorporated strategies improve performance respectively. In section IV, the performance of LSEDA-gl is analyzed by 7 test functions of 100, 500, 1000 D provided in CEC'2008 special session on LSGO and then, summarizes the experimental results and discusses the strengths and weaknesses of LSEDA-gl. In section V, the main features of LSEDA-gl are concluded.

II. PRINCIPLES OF LSEDA-GL

The classical EDAs have been proven to be effective on most classical test functions with less than 100 D. However, the Ds of practical engineering and scientific optimization problems are usually very large. Similar to other EAs, EDA often suffers from the curse of dimensionality, which implies that its performance deteriorates quickly as the dimensionality of search space increases [1]. Several strategies, such as cooperative co-evolution in which only a subgroup of variables are optimized in cooperative co-evolution in one iteration, were proposed to make GA [8] and DE [9] more effective for high dimensional problems. However, our study focuses on solving LSGO problems by EDA, all of the variables are optimized quickly and synchronously during the optimization process.

A. Framework of Estimation of Distribution Algorithm for Large Scale Optimization

To reduce the complexity of conducting the probabilistic model, the univariate EDA whose variables are considered independently is adopted in the framework of our algorithm. Similar to UMDAc, the joint probability distribution over a set of random variables $x = \{x_i\}$ where $i = 1, 2, ..., D$ for D dimensional space is defined as follows:

$$P(\vec{x}) = \prod_{i=1}^{M} P(x_i);$$ (1)

The probability distribution used to model each variable $P(x_i)$ is a single Gaussian distribution defined as follows:

$$N(\mu, \nu) = \frac{1}{(2\pi\nu^2)^{1/2}} e^{(-\frac{|x-\mu|^2}{2\nu^2})}.$$ (2)

In this formula, $N(\mu, \nu)$ stands for a Gaussian distribution with mean μ and standard deviation ν. In contrast to IDEA [10] developed by Bosman which requires computing all elements of covariance matrix to adapt an arbitrary Gaussian, LSEDA-gl abandons adapting the non-diagonal elements in covariance matrix, which reduces the computational cost remarkably for LSGO problems.

Unlike some other EDAs, the Gaussian probability model $P(\vec{x})$ for LSEDA-gl is built by the current population only. Given current population S_{parent} including NP individuals, a subpopulation S_{sel} including N individuals $X_1, X_2, ..., X_N$ ($N < NP$) with top fitness values are selected to update the model for the next generation. Then, the mean vector $\overline{X}$ and

the standard deviation vector δ are generated by eq.(3) and eq.(4) respectively:

$$\overline{X} = \frac{1}{N} \sum_{i=1}^{N} X_i,$$ (3)

$$\delta = \{Var(x_1), Var(x_2), ..., Var(x_m)\},$$ (4)

where N is the size of the selected subpopulation, and X_i stands for the ith individual of the selected subpopulation. The Var operator is to determine the variance of each variable to form the standard deviation vector δ. The variances of population X are verified in the following formula:

$$Var(x_i) = \frac{1}{N-1} \sum_{j=1}^{N} (X_{ji} - \overline{X}_i)^2,$$ (5)

X_{ji} stands for the ith element of the jth individual and $\overline{X}_i$ denotes the ith element of the mean vector $\overline{X}$. The new population S_{parent} of the next generation are sampled under a mixed Gaussian and lévy distribution. For LSEDA-gl, selected proportion P_{sel} is used to truncate the best $(P_{sel} \cdot NP)$, e.g. N, population. Through a large number of experiments, it is observed that the selected proportion is not sensitive in most cases. The procedure of LSEDA-gl is described as follows and the details of the SGLD, STDC and restart strategies will be introduced in the following subsections.

Framework of LSEDA-gl:

input:
- LSGO problem;
- weight vector W (for SGLD);
- a stopping criterion (for restart);
- a termination condition;
- population size NP and selected population size N.

output: global optimum.
- **Step1** Randomly initialize S_{parent} including NP individuals and then, generate the Gaussian probability model $P(\vec{x})$, which is defined by the mean vector $\overline{X}$ and the standard deviation vector δ in typical single Gaussian probability distribution model, by S_{parent};
- **Step2** Create the new population X based on the mixed Gaussian and Lévy distribution probability model $P'(\vec{x})$ (SGLD);
- **Step3** Evaluate the objective function $f(X)$ for each solution X_i in population X of which the population size is NP: $F = f(X_1), ..., f(X_{NP})$;
- **Step4** Update $P(x)$ with the best (N) individuals selected to estimate the current distribution. Where N is the selected population size set beforehand, and NP is the size of the whole population;
- **Step5** Standard deviation control strategy (STDC);
- **Step6** If the stopping criterion is met, restart strategy is adopted;
- **Step7** Go to step 2 (until termination condition).

B. Sampling under mixed Gaussian and Lévy Distribution

Definition: Consider a process represented by a set Y_i of identically distributed random variables. If the sum of these random variables has the same probability distribution as

2008 IEEE Congress on Evolutionary Computation (CEC 2008)

individual random variables, then this process is called ***stable*** [7].

The Gaussian process is a typical example of stable process with finite second moment which would lead to a characteristic scale and the Gaussian behavior for the probability density through the central limit theorem [7]. It is acknowledged that Gaussian probability distribution plays an important role in modelling the structures of the continuous optimization problems and sampling the solutions for the next generation. Different from Gaussian probability distribution whose variance can be denoted as a finite scalar, a class of probability distributions with an infinite second moment that also yields a stable process are discovered by P. Lévy in the 1930s [5] [6]. The formal representation for such class of probability can be respected as follows [5] [6]:

$$\text{Ł}_{\alpha,\gamma} y = \frac{1}{\pi} \int_0^\infty e^{-\gamma q^\alpha} \cos(qy) dq \quad y \in \mathbf{R}, \quad (6)$$

where γ is the scaling factor satisfying $\gamma > 0$, and α controls the shape of the distribution, requiring $0 < \alpha < 2$. More analytic details about the Lévy distribution are available in [5] [6] and [7]. The Cauchy probability distribution which is adopted in FEP, is a special case of the Lévy probability distribution with $\alpha = 1$. For the limit of $\alpha = 2$, the distribution is reduced to be the classical Gaussian distribution which is not included in Lévy probability distribution class.
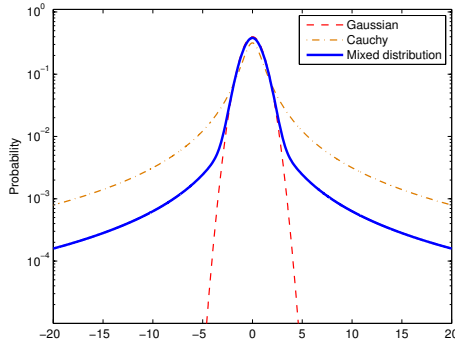

Fig. 1. Comparison among mixed Lévy $\alpha = 1$ and Gaussian distribution, Lévy probability distribution with $\alpha = 1$ and Gaussian probability distribution. The vertical coordinate is of log scale and $\gamma = 1$.

Although the analytic form of the integral is still unknown for general α, it has been known for the rules of changes in the sharps with different α: the smaller is the parameter α, the longer is the tail. The characteristic of infinite second moment in Lévy probability provides a much wider distribution. LSEDA mix Gaussian distribution with Lévy distribution ($\alpha = 1$, $\gamma = 1$) to form a new probability distribution N_L shown as eq.(7) for sampling offspring of the next generation.

$$randnum = rand;$$
$$P_m = \frac{10 \cdot randnum}{D}$$

$$N_L = \begin{cases} 0.9 \cdot N(0,1) + 0.1 \cdot L_{1,1}, & if\ D < 100 \\ (1 - P_m) \cdot N(0,1) + P_m \cdot L_{1,1}, & otherwise, \end{cases} \quad (7)$$

where $N(0,1)$ stands for the Gaussian distribution with mean 0 and standard deviation 1; $L_{1,1}$ is the Lévy probability with $\alpha = 1$, $\gamma = 1$; D is the dimensional size of the problem; $randnum$ is generated randomly based on uniform distribution between $(0,1)$. The probability distribution for creating each offspring on different dimensional size is self-adapted as shown in eq.(7): a smaller mixture probability P_m is likely adopted for a larger dimensional size. A comparison of the sharps among the most bias mixed Gaussian and Lévy $\alpha = 1$ distribution, Lévy probability distribution with $\alpha = 1$ and Gaussian probability distribution is shown in Fig.1. It is obvious that the sharp of the mixed distribution in the area near the mean is similar to the Gaussian distribution which is able to maintain the previous model in a great measure. Taking advantage of the characteristic of Lévy distribution, the mixed distribution is also equipped with an infinite second moment. It is beneficial when the global optimum is sufficiently far away from the current search point and therefore, speed up the convergence rate. Moreover, it remarkably reduces the probability of the model being trapped in some of the local optima.

The formal way of sampling operator in LSEDA-gl is defined as follows:

$$X_{ji} = \overline{X}_i + \delta_i^{\frac{1}{2}} \cdot N_L, \quad (8)$$

where $\overline{X}_i$ is the ith element of the mean vector $\overline{X}$. In eq.(8), the ith element of jth individual is sampled under mixed Gaussian and Lévy distribution model N_L. Compared with UMDAc, whose sampling operator is denoted as $X_{ji} = \overline{X}_i + \delta_i^{\frac{1}{2}} \cdot N(0,1)$, the standard Gaussian probability $N(0,1)$ is replaced with the mixed Gaussian and Lévy distribution N_L.

C. Standard Deviation Control Strategy

The main idea of STDC strategy is to estimate a common threshold of standard deviations for all variables at every iteration of the optimization process to control their shrinking speeds and therefore, to control the decreasing speed of diversity. The variables that have lower standard deviation values than the corresponding thresholds will be forced to set their standard deviations to be the corresponding thresholds. The weighted mean of the standard deviations of all variables is used as a control standard here. The details of STDC are shown as follows:

Pseudo code of STDC
---
for j=1:D // D stands for the size of the dimensions
   if $\delta_j < W(j) \times \overline{\delta}$
      $\delta_j = W(j) \times \overline{\delta}$
      /* $\overline{\delta}$ is the mean of the whole dimensions */
      /* W is a weight vector set beforehand */
   end if
end for

W is a weight vector set beforehand. For simple problems , (e.g., Sphere function and Ackley function), $W(i) = 0$,

$i = 1, ... D$ are proper for the algorithm to converge to the global optimum quickly, while $W(i) = 1, i = 1, ... D$ are also effective and steadier for such simple problems although a little longer time is needed and the accuracy is a bit worse. $\bar{\delta}$ is the mean of standard deviations of the whole variables. The application of STDC provides a much better performance for hard problems, such as Schwefel function, Rosenbrock function and Rastrigin function.

Under a great deal of experimental experiences, the advantages of STDC may be summarized into two aspects: 1) STDC strategy may control the decreasing speed of population diversity adaptively via controlling the shrinking speed of standard deviation of Gaussian variables; 2) By sharing the common threshold of standard deviation, which is the weighted mean of standard deviation of all variables, the STDC strategy makes it possible that the variation of each variable is controlled (or influenced) by states of all other variables. This enables the communication among variables.

After analyzing a great deal of W values for each D, LSEDA-gl applies a self-adapted setting of initial W_0 value for problems ofdifferent Ds as shown in eq.(9):

$$W_0 = 0.55 - e^{lg(\frac{D}{10^5})} \tag{9}$$

In formula eq.(9), the metric W_0 is determined by D only, that means a larger W_0 is adopted for problems of lower D. $W = W_0$ and $W = 0$ are adopted by turns in LSEDA-gl. After one restart, the value W changes to other one. The formal procedure of STDC is presented as follows:

Rules of change W value
Initialize $W = W_0$ by eq.(9).
if a restart happens /*W value changes in each restart*/
if $W == W_0$
$W = 0$
else if $W == 0$
$W = W_0$
end if
end if
end

D. Restart Strategy

For the restart strategy, the LSEDA is stopped whenever stopping criterion described below is met, and a restart is launched with the model of which the mean vector is the same as the previous one and the standard deviation vector is set to be initial standard deviation decreased by a factor of 2.

To decide when to restart, the following criteria are used:

1.Stop if the best objection function value obtained keeps unchanged for 100 generation;

2.Stop if the previous mean of standard deviations of all variables is more than the double of that 100 generation before.

The convergence graphs for worst run by LSEDA on 4 test functions provided in CEC'2008 special session are shown in Fig.2 and Fig.3. With restart strategy, the LSEDA is able to accomplish more than one searches within the fixed
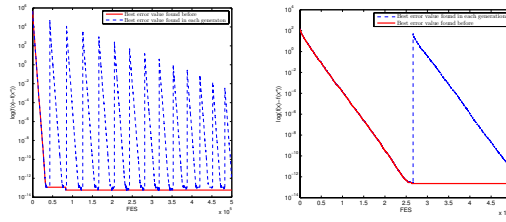

Fig. 2. Convergence graph for worst run of LSEDA for sphere function on the left and schwefel function on the right
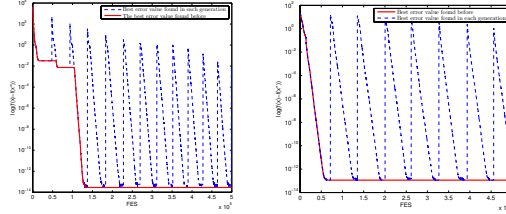

Fig. 3. Convergence graph for worst run of LSEDA for Griewank function on the left and Ackley function on the right

maximum number of fitness evaluation (FES). Particularly, the performance of LSEDA is improved significantly on shift Griewank function.

III. COMPARISON AMONG LSEDA-GL, EDA-STDC AND UMDAC ON CLASSICAL BENCHMARK FUNCTIONS

To provide experimental evidence to study how the incorporated strategies SGLD and STDC improve the performance respectively, we compare LSEDA-gl, EDA-STDC and UMDAc on the first experimental group that comprises 7 classical benchmark problems taken from the standard literature with D scaling from 100 to 200, which is also adopted in [19]. The formal definition of the benchmark functions are summarized in Table I. Function 1-4 are unimodal functions, and function 5-7 are multimodal functions. The average experimental results of 50 runs are summarized in Table II. In Table II, the best result for each function is written in boldface. Based on the above introduction, SGLD makes it possible to adopt small NP to handle a high dimensional problem while STDC reduces the metric P_{sel} sufficiently. Therefore, we set $NP = 100$ for LSEDA-gl while $NP = 500$ for the other algorithms; set $P_{sel} = 0.4$ for UMDAc while $P_{sel} = 0.2$ for the other algorithms. Without general, FES=$1000 \times D$ is adopted for all functions.

For unimodal functions 1-3, LSEDA-gl provides the most superior results within fixed FES among all of the considered algorithms, while either of the other two algorithms could obtain a comparable result on any one function. For the dimensional size scaling to 200, LSEDA-gl shows a steady performance that no significant unstable indication is apparent. For discontinuous problem function 4, UMDAc without STDC or SGLD fails to converge to global optimum in all runs. LSEDA-gl provides the worst result on function 5,

3920   *2008 IEEE Congress on Evolutionary Computation (CEC 2008)*

Num	Problems	Bounds	Objective function	Global location	Global optimum
fun1	**Sphere**	[-100,100]	$f_1(x) = \sum_{i=1}^n x_i^2$	$x_i = 0$	0
fun2	**Schwefel**	[-10,10]	$f_2(x) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|$	$x_i = 0$	0
fun3		[-100,100]	$f_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j^2)$	$x_i = 0$	0
fun4	**Step**	[-100,100]	$f_4(x) = \sum_{i=1}^n | x_i + 0.5 |^2$	$x_i = -0.5$	0
fun5	**Rastrigin**	[-5.12, 5.12]	$f_5(x) = \sum_{i=1}^n (x_i^2) - 10cos(2\pi x_i) + 10$	$x_i = 0$	0
fun6	**Ackley**	[-32,32]	$f_6(x) = -20 \cdot exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2})$ $+e - exp(\frac{1}{n}\sum_{i=1}^n cos(2\pi x_i)) + 20$	$x_i = 0$	0
fun7	**Griewank**	[-600,600]	$f_7(x) = \frac{1}{4000}\sum_{i=1}^n x_i^2 + 1 - \prod_{i=1}^n cos(\frac{x_i}{\sqrt{i}})$	$x_i = 0$	0

Algorithms	Metric	fun1	fun2	fun3	fun4	fun5	fun6	fun7
				D=100				
LSEDA-gl	mean	**3.2684E-35**	**2.6528E-17**	**1.0189E-31**	**0.0000E+00**	1.0887E+03	**1.1546E-14**	**0.0000E+00**
1E5 FES	std	**2.1599E-35**	**2.0777E-17**	**1.5799E-31**	**0.0000E+00**	2.9026E+01	**0.0000E+00**	**0.0000E+00**
EDA-STDC	mean	4.1389E-08	1.9383E-06	1.3435E-11	**0.0000E+00**	9.3562E+00	5.0662E-06	2.0319E-13
1E5 FES	std	1.3088E-07	2.3480E-07	2.1947E-12	**0.0000E+00**	**2.7035E+00**	1.5713E-05	4.1959E-14
UMDAc	mean	1.1731E-05	7.6133E-03	4.7472E-04	**0.0000E+00**	2.4011E+02	5.3620E-04	7.1796E-06
1E5 FES	std	1.2481E-06	5.1813E-04	6.9361E-05	**0.0000E+00**	4.3914E+01	3.3068E-05	1.2193E-06
				D=200				
LSEDA-gl	mean	**9.7380E-44**	**5.7504E-24**	**9.9185E-33**	**0.0000E+00**	3.8401E+02	**2.2204E-14**	**0.0000E+00**
2E5 FES	std	**8.4729E-44**	**5.2535E-24**	**3.1363E-32**	**0.0000E+00**	3.9718E+01	**3.3495E-15**	**0.0000E+00**
EDA-STDC	mean	1.7872E-19	2.4482E-09	1.5686E-17	**0.0000E+00**	**3.5637E+01**	4.8374E-11	**0.0000E+00**
2E5 FES	std	1.7082E-20	2.5145E-10	5.5739E-18	**0.0000E+00**	**6.9373E+00**	3.2422E-12	**0.0000E+00**
UMDAc	mean	3.0427E-09	2.1107E-04	2.5604E-07	3.1380E+00	1.4737E+02	6.7739E-01	1.0093E-09
2E5 FES	std	3.7945E-10	1.1624E-05	3.0173E-08	1.2172E+00	3.9309E+02	2.1421E+00	1.4219E-10

the reason may be summarized as follows: SGLD strategy prevents the whole population from being trapped by any of the local optima as other algorithms which lowers the accuracy of the last result. For the other two multimodal problems function 6 and 7, LSEDA-gl is able to solve them kindly. In summary, the reasons of implementing multiform strategies are summarized explicitly:

- 1) STDC mainly focuses on controlling the decreasing speed of population diversity and enabling the communication among variables.
- 2) SGLD makes it possible to accelerate the convergence speed remarkably, and it also exhibits the similar effect as STDC on escaping from the local optima.

IV. EXPERIMENT FOR CEC'08 COMPETITION ON LSGO

A. Configuration

The benchmark suite for experiment consists of 7 test functions defined in [1]. Functions 1 and Function 2 are unimodal functions and Functions 3- 6 are multimodal functions. To prevent exploitation of symmetry of the search space and of the typical zero value associated with the global optimum, the local optima of classical functions are shifted to a value different from zero and the function values of the global optima are non-zero [2]. Function 7 is a complicated problem of which the global optima is unknown so far.

The FES is defined by $5000 \times D$, where D is the dimensional size of the problem. The error value recorded finally is the absolute margin between the fitness of the best solution found and the fitness of the global optimum. Experimental results of 25 independent runs for each problems with 100, 500 and 1000 D are recorded and storage in the appendix Tables III - V following the format required. The initial population were randomly generated within the search space and the value of variables sampled outside the search space during the evolution process were reset to the corresponding boundaries. Experiments were conducted by LSEDA-gl presented above. The hardware was workstation (Inter(R) Core(TM) 2.0 GHZ, Quad CPU Q 6600, @ 2.4GHZ, 3.93GB RAM) running Windows.

B. Parameters setting

To obtain better experimental results, the parameters such as population size NP and selected population N can be tuned to suit each problem. However, the structure of practical science and engineering problems are usually unknown beforehand. In order to make our algorithm general for most

problems, we choose parameters as consistently as possible.

It is obvious that for an easy problem (e.g., unimodel problems and low dimensional problems), a small value of NP is sufficient, but for difficult problems, a large value of NP is recommended in order to avoid stagnation to a local optimum. Benefit from these efficient strategies SGLD, STDC and restart strategy, a much smaller population size is sufficient to exploit structural information of high dimensional search space efficiently compared to classical EDAs. For conducting the experimental results, we set $NP = 50$, $N = 10$ for 100 dimension problems, set $NP = 100$, $N = 20$ for 500 dimension problems and set $NP = 200$, $N = 30$ for 1000 dimension problems.

C. Results and Discussion

Experimental results conducted by LSEDA-gl are presented in the Appendix. Tables II-V show the 1st best, the 7th best, the 13th (median), the 19th best, the 25th (worst), mean, and standard deviation values of 25 independent runs under fixed max FES. Fig.4, show the convergence graphs for functions 1 - 7 with 1000 D.

According to the results, LSEDA-gl is able to reach the accuracy level (the error value is lower than 1.0E-8, which is also adopted in [2]) with one tenth of the given FES on Function 1, Function5 and Function6 of all dimensional size. For Function 2 shift Schwefel problem whose optimization difficulty arises sharply as D increase, LSEDA-gl provides a steady convergence speed towards global optimum on all dimensional setting, although completely convergence has not been accomplished within the fixed FES. The results achieved on Function 3 - 4 are dissatisfied, error values were two orders of magnitude for 100D and more for 500 D and 1000 D. The reason may be that LSEDA-gl is a single Gaussian probability model based algorithm, so the simple model makes this algorithm more effective for unimodal functions. For multimodal functions, Shifted Griewank Function and Shifted Ackley Function can still be solved kindly. However, the LSEDA-gl can't explore the structure of Rosenbrock fully, of which the valley from local optimum to global optimum is very narrow. All runs will be trapped by the huge local optima of Rastrigin problem. These strategies are not sufficient enough to force the search area close enough to the global optima on these high dimensional problems.

V. CONCLUSION

A univariate EDA LSEDA-gl is proposed to extend EDA to high dimensional problems in this paper. Compared to classical UMDAc, the major advantage of LSEDA-gl is to preserve a reasonably wider search scope during the optimization process. The reason may be that three major strategies SGLD, STDC and restart strategy are able to control the decreasing speed of population diversity adaptively via controlling the shrinking speed of standard deviation of Gaussian variables. Experimental results on classical functions show the advantages of SGLD and STDC respectively. Moreover, results of LSEDA-gl for the CEC'2008 Special session on LSGO

have been presented above. According to the results, the performances of LSEDA-gl are steady for most problems.

Balancing the relationship between optimization and learning for EDAs is quite a difficult task because precision of learning may delay the convergence speed or get caught up in some of the local optima. Compared to the previous version EDA-STDC [18], LSEDA-gl reduces the convergence time significantly. Particularly, our SGLD strategy could be simply incorporated into existing EDAs to accelerate the convergence speed and escape from the local optima.

REFERENCES

[1] K. Tang, X. Yao, P. N. Suganthan, C. MacNish, Y. P. Chen, C. M. Chen, Z. Yang, "Benchmark Functions for the CEC2008 Special Session and Competition on Large Scale Global Optimization", Technical Report for CEC'2008 special session, Nov. 2007.

[2] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, S. Tiwari, "Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization", Technical Report for CEC'2005 special session, May. 2005.

[3] X. Yao, An overview of evolutionary computation, *Chinese J. Adv. Software Res.,* vol. 3, no. 1, pp. 12-29, 1996.

[4] Q. Lu and X. Yao, "Clustering and Learning Gaussian Distribution for Continuous Optimization", *IEEE Trans. Systems*, *Man*, and *Cybernetics*, vol. 35, no. 2, pp. 195-204, May. 2005.

[5] P. Lévy, *Theorie de l'Addition des Veriables Aleatoires.* Paris, France: Gauthier-Villars, 1937.

[6] B. Gnedenko and A. Kolmogorov, *Limit Distribution for Sums of Independent Random Variables.* Cambridge, MA: Addition-Wesley, 1954.

[7] Ch-Y. Lee, and X. Yao, "Evolutionary Programming Using Mutations Based on the Lvy Probability Distribution", *IEEE Trans. Evol.Comput.*, vol. 8, no. 1, pp. 1-13, Feb. 2004.

[8] A. M. Potter and K. A. De Jong, "A cooperative co-evolutionary approach to function optimization", *Proc. the Third International Conference on Parallel Problem Solving from Nature*, pp. 249-257. Springer-Verlag, 1994.

[9] Z. Yang, K. Tang and X. Yao "Differential Evolution for High-Dimensional Function Optimization", *Proc. IEEE Congress on Evolutionary Computation*, Singapore, 2007, pp. 3523-3530.

[10] P. A. N. Bosman, Design and application of iterated density-estimation evolutionary algorithms, Ph.D. dissertation, Utrecht Univ., TB Utrecht, The Netherlands, 2003.

[11] J. Boyan A. J. Buntine, "Statistical machine learning for large-scale optimization", *Neural Computing Surveys*, 3:1-58.

[12] P. Larrañga "A review on estimation of distribution algorithms", in *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, chapter 3, pp. 57-100. Kluwer.

[13] M. Gallagher, M. Frean "Population-Based Continuous Optimization, Probabilistic Modelling and Mean Shift", *MIT press on Evolutionary Computation*, vol. 13 no. 1, pp. 29-42.

[14] B. Yuan, Marcus Gallagher "Experimental Results for the Special Session on Real-Parameter Optimization at CEC 2005: A Simple, Continuous EDA", *Proc. Conference on Evolutionary Computation*, special issue on Real-Parameter Optimization, 2005.

[15] M. Sebag and A. Ducoulombier, "Extending population-based incremental learning to continuous search spaces", *Parallel Problem Solving from Nature-PPSN VI*, 1998, pp. 418-427.

[16] S. Rudlof and M.Köppen, "Stochastic hill climbing by vectors of normal distributions, *Proc. 1st OnlineWorkshop on Soft Computing (WSC1)*, Nagoya, Japan, 1996.

[17] P. Larrañga, R. Etxeberria, J. A. Lozano, and J. M. Peña, "Optimization in continuous domains by learning and simulation of Gaussian networks," *Proc. 2000 Genetic and Evolutionary Computation Conf. Workshop Program*, pp. 201-204, 2000.

[18] Y. Wang, B. Li, "Estimation of Distribution Algorithm for Large Scalar Global Optimization", submitted for CEC'2008 special issue on "Large Scalar Global Optimization".

[19] Z. Yang, J. He, X. Yao, Make a difference to differential evolution, submitted to a book chapter, 2006.

TABLE III

ERROR VALUES ACHIEVED FOR PROBLEMS 1-7, WITH D=100

fun	fun1	fun2	fun3	fun4	fun5	fun6	fun7
D=100 FES=5E3							
1st	1.0122E+02	3.6016E+01	2.1437E+06	3.4703E+02	2.2446E+00	3.4733E+00	-8.4326E+02
7th	1.8363E+02	4.0512E+01	3.7601E+06	4.8406E+02	2.7479E+00	3.8003E+00	-8.1056E+02
13th	2.3501E+02	4.4353E+01	5.4631E+06	5.2039E+02	2.9647E+00	3.9725E+00	-8.0563E+02
19th	2.7841E+02	4.7180E+01	8.4354E+06	5.4399E+02	3.5713E+00	4.2279E+00	-7.9178E+02
25th	4.1218E+02	5.9915E+01	1.8838E+07	5.7664E+02	5.4630E+00	4.7065E+00	-7.7887E+02
mean	2.3646E+02	4.4466E+01	6.7034E+06	5.0693E+02	3.2183E+00	4.0313E+00	-8.0432E+02
std	7.6869E+01	5.7252E+00	3.7500E+06	5.4128E+01	7.8441E-01	3.3058E-01	1.6930E+01
D=100 FES=5E4							
1st	5.6843E-14	6.9254E-02	1.7717E+02	6.4672E+01	2.8422E-14	6.5370E-13	-1.4691E+03
7th	5.6843E-14	8.9067E-02	2.7404E+02	8.8552E+01	2.8422E-14	1.1369E-12	-1.4175E+03
13th	5.6843E-14	1.0481E-01	1.3085E+03	9.8451E+01	2.8422E-14	1.5632E-12	-1.4040E+03
19th	5.6843E-14	1.2303E-01	5.4224E+03	1.1443E+02	2.8422E-14	3.4390E-12	-1.3937E+03
25th	1.1369E-13	1.6408E-01	1.0835E+04	1.8406E+02	1.2321E-02	1.0900E-10	-1.3822E+03
mean	6.8212E-14	1.0526E-01	2.8947E+03	1.0353E+02	1.2815E-03	6.5779E-12	-1.4179E+03
std	2.3206E-14	2.3710E-02	3.6931E+03	2.4945E+01	3.6358E-03	2.1421E-11	2.4774E+01
D=100 FES=5E5							
1st	5.6843E-14	1.7053E-13	8.2525E+01	6.4672E+01	2.8422E-14	8.5265E-14	-1.4896E+03
7th	5.6843E-14	2.2737E-13	9.3542E+01	8.8552E+01	2.8422E-14	8.5265E-14	-1.4740E+03
13th	5.6843E-14	2.2737E-13	1.2934E+02	9.8451E+01	2.8422E-14	8.5265E-14	-1.4672E+03
19th	5.6843E-14	2.2737E-13	1.6760E+02	1.1443E+02	2.8422E-14	1.1369E-13	-1.4601E+03
25th	5.6843E-14	2.8422E-13	1.4651E+03	1.8406E+02	2.8422E-14	1.1369E-13	-1.4433E+03
mean	5.6843E-14	2.2055E-13	2.8121E+02	1.0353E+02	2.8422E-14	9.7771E-14	-1.4638E+03
std	3.8646E-29	2.4994E-14	4.2642E+02	2.4945E+01	1.9323E-29	1.4399E-14	1.4513E+01

TABLE IV

ERROR VALUES ACHIEVED FOR PROBLEMS 1-7, WITH D=500

fun	fun1	fun2	fun3	fun4	fun5	fun6	fun7
D=500 FES=2.5E4							
1st	1.3826E+03	7.5339E+01	7.2480E+07	2.0964E+03	1.0688E+01	3.8726E+00	-5.1775E+03
7th	1.4794E+03	7.9429E+01	8.8098E+07	2.1108E+03	1.2981E+01	4.0073E+00	-5.1463E+03
13th	1.5868E+03	7.9866E+01	9.0523E+07	2.2141E+03	1.5213E+01	4.1157E+00	-5.1196E+03
19th	2.0062E+03	8.0884E+01	9.6228E+07	2.2668E+03	1.7053E+01	4.2178E+00	-5.1081E+03
25th	2.2616E+03	8.1648E+01	1.1577E+08	2.4284E+03	1.7677E+01	4.3164E+00	-4.9073E+03
mean	1.7315E+03	7.9566E+01	9.2665E+07	2.2228E+03	1.4955E+01	4.1082E+00	-5.1031E+03
std	3.3399E+02	2.0127E+00	1.5105E+07	1.1279E+02	2.4196E+00	1.4815E-01	8.2338E+01
D=500 FES=2.5E5							
1st	2.2737E-13	4.8584E+00	1.1800E+03	7.7829E+02	1.1369E-13	4.7180E-12	-6.8512E+03
7th	2.2737E-13	5.0849E+00	1.2478E+03	8.2770E+02	1.4211E-13	8.3560E-12	-6.8290E+03
13th	2.8422E-13	5.3242E+00	1.3512E+03	8.6489E+02	1.4211E-13	1.0260E-11	-6.7963E+03
19th	2.8422E-13	5.6366E+00	1.6718E+03	8.8901E+02	1.4655E-08	2.2595E-11	-6.7438E+03
25th	2.8422E-13	6.0216E+00	1.9853E+03	9.2250E+02	9.8573E-03	1.2997E-10	-6.7382E+03
mean	2.6527E-13	5.3571E+00	1.4391E+03	8.5906E+02	1.9170E-03	2.5744E-11	-6.7917E+03
std	2.8422E-14	4.2051E-01	2.8073E+02	5.0120E+01	3.8534E-03	3.9651E-11	4.5021E+01
D=500 FES=2.5E6							
1st	2.2737E-13	1.9588E-10	6.2806E+02	7.7788E+02	1.1369E-13	3.1264E-13	-6.8952E+03
7th	2.2737E-13	2.4204E-10	7.6352E+02	8.2634E+02	1.1369E-13	3.1264E-13	-6.8659E+03
13th	2.2737E-13	2.6461E-10	8.5068E+02	8.5988E+02	1.1369E-13	3.1264E-13	-6.8262E+03
19th	2.2737E-13	3.1292E-10	9.2960E+02	8.8117E+02	1.1369E-13	3.1264E-13	-6.7792E+03
25th	2.2737E-13	3.2782E-10	1.1578E+03	9.2160E+02	1.1369E-13	3.1264E-13	-6.7672E+03
mean	2.2737E-13	2.7152E-10	8.6745E+02	8.5569E+02	1.1369E-13	3.1264E-13	-6.8276E+03
std	0.0000E+00	4.3927E-11	1.8731E+02	5.0784E+01	0.0000E+00	5.3550E-29	4.8356E+01

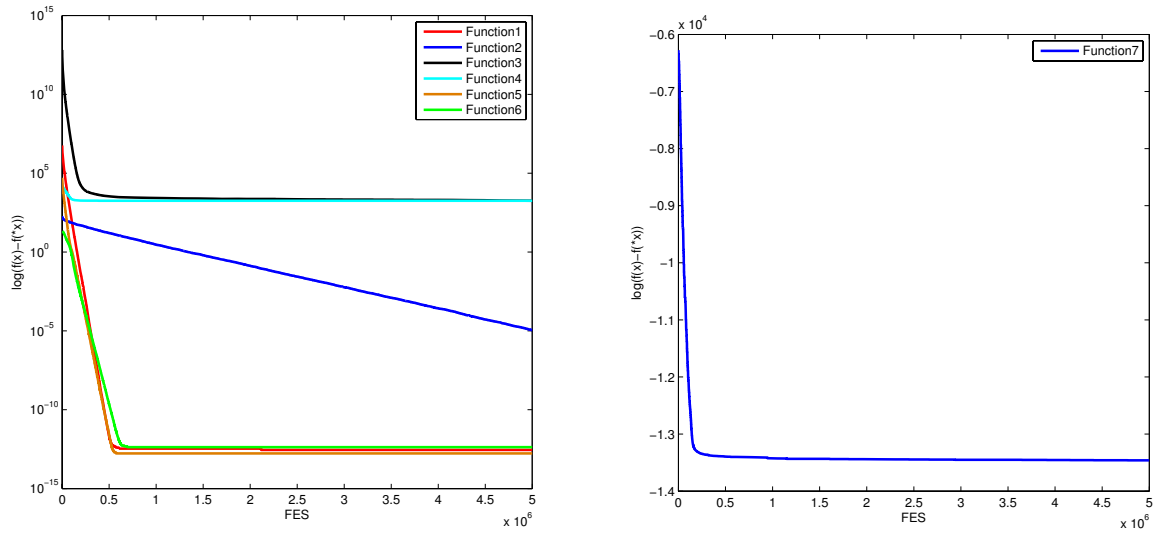*2008 IEEE Congress on Evolutionary Computation (CEC 2008)*   3923

Fig. 4. Convergence Graph for Functions 1-6 (D=1000) on the left and Function 7 (D=1000) on the right

TABLE V

ERROR VALUES ACHIEVED FOR PROBLEMS 1-7, WITH D=1000

fun	fun1	fun2	fun3	fun4	fun5	fun6	fun7
D=1000 FES=5E4							
1st	7.9782E+03	9.0491E+01	7.3785E+08	1.3377E+04	7.8257E+01	5.0921E+00	-9.8576E+03
7th	9.5240E+03	9.2269E+01	8.4886E+08	1.3515E+04	8.5312E+01	5.3769E+00	-9.7907E+03
13th	9.7459E+03	9.3182E+01	8.9688E+08	1.3571E+04	8.5907E+01	5.5497E+00	-9.6350E+03
19th	1.0792E+04	9.3441E+01	9.7637E+08	1.3606E+04	9.3398E+01	5.6247E+00	-9.5717E+03
25th	1.1641E+04	9.4133E+01	1.0641E+09	1.3743E+04	9.8897E+01	5.6792E+00	-9.4226E+03
mean	9.9496E+03	9.2770E+01	9.0548E+08	1.3572E+04	8.8254E+01	5.4650E+00	-9.6609E+03
std	1.0753E+03	1.1032E+00	9.8140E+07	9.3528E+01	7.9653E+00	2.1064E-01	1.4290E+02
D=1000 FES=5E5							
1st	9.0949E-13	1.4794E+01	3.2512E+03	4.1730E+03	2.6432E-12	2.2746E-10	-1.3523E+04
7th	1.3074E-12	1.5004E+01	3.4149E+03	4.3835E+03	2.7285E-12	3.0505E-10	-1.3442E+04
13th	2.0464E-12	1.5570E+01	3.6160E+03	4.4663E+03	3.3822E-12	3.1912E-10	-1.3395E+04
19th	2.4443E-12	1.6470E+01	3.7164E+03	4.5557E+03	3.6380E-12	1.0010E-09	-1.3379E+04
25th	1.8922E-08	1.7482E+01	3.9721E+03	4.7459E+03	3.4362E-02	1.3253E-04	-1.3306E+04
mean	2.4691E-09	1.5888E+01	3.5853E+03	4.4866E+03	1.5724E-03	2.2890E-05	-1.3408E+04
std	6.2647E-09	1.0057E+00	2.3304E+02	1.3556E+02	1.5367E-02	4.7739E-05	6.1671E+01
D=1000 FES=5E6							
1st	2.8422E-13	9.7908E-06	1.5707E+03	5.0549E+02	1.7053E-13	4.2633E-13	-1.3590E+04
7th	2.8422E-13	9.8559E-06	1.6403E+03	5.3529E+02	1.7053E-13	4.2633E-13	-1.3510E+04
13th	3.4106E-13	1.0435E-05	1.6755E+03	5.4532E+02	1.7053E-13	4.2633E-13	-1.3461E+04
19th	3.4106E-13	1.0552E-05	1.7979E+03	5.5618E+02	1.7053E-13	4.2633E-13	-1.3452E+04
25th	3.4106E-13	1.1178E-05	2.0059E+03	5.8106E+02	1.7053E-13	4.2633E-13	-1.3391E+04
mean	3.2211E-13	1.0394E-05	1.7288E+03	5.4493E+02	1.7053E-13	4.2633E-13	-1.3481E+04
std	2.8422E-14	5.1087E-07	1.4017E+02	1.7966E+01	0.0000E+00	0.0000E+00	5.7186E+01