A hybrid EDA with Chaotic DE algorithm and its performance analysis

Article *in* Journal of Computational Information Systems · February 2015

DOI: 10.12733/jcis13564

Fuqing Zhao
Lanzhou University of Technology

Zhongshi Shao
Shaanxi Normal University

Chun (Chuck) Zhang
Georgia Institute of Technology

Journal of Computational Information Systems 11: 4 (2015) 1505–1512
Available at http://www.Jofcis.com

A Hybrid EDA with Chaotic DE Algorithm and Its Performance Analysis ⋆

Fuqing ZHAO [1,4], Zhongshi SHAO [1,∗], Rundong WANG [2], Chuck ZHANG [3], Junbiao WANG [4]

[1] *School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China*

[2] *China Machinery Engineering Corporation, Beijing 100055, China*

[3] *H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA*

[4] *Key Laboratory of Contemporary Design, Integrated Manufacturing Technology, Ministry of Education, Northwestern Polytechnical University, 710072, China*

Abstract

This paper presents a novel hybrid algorithm termed as a hybrid EDA with chaotic DE algorithm (EDcDE), which is designed to effectively combine estimation of distribution algorithm (EDA) and differential evolution algorithm (DE) without losing their unique features. Due to its effective nature of harmonizing the global search of EDA with the local search of DE, and chaotic policy which is introduced to DE to strengthen the search ability of DE, EDcDE can discover the optimum solution in a rapid and reliable way. Meanwhile the global convergence of the proposed algorithm was analyzed with the theory of stochastic processes. The proposed algorithm was used to solve global optimization problems. The experimental results on benchmark functions have demonstrated the effectiveness of the proposed approach.

Keywords: Hybrid Optimization; Estimation of Distribution; Differential Evolution; Convergence Analysis; Global Optimization

1 Introduction

Estimation of distribution algorithms (EDAs) is a class of evolutionary algorithms characterized by the use of probability models. In EDAs, new candidate individuals of next iteration are not reproduced through crossover or mutation operator of traditional genetic algorithms, but generated by sampling from a probability distribution of promising individuals [1]. However, EDAs has two cruxes. First of all, EDAs may cause the problem of overfitting the search space and cannot represent the general information. Then due to the premature convergence of EDAs,

⋆Project supported by the National Nature Science Foundation of China (No. 51365030).

∗Corresponding author.

Email address: `shaozhongshi@hotmail.com` (Zhongshi SHAO).

DOI: 10.12733/jcis13564
February 15, 2015

the probability models no longer generate diversified solutions resulted in poor performance [2]. Therefore, many researchers have already done much research in combining EDAs with other optimum algorithms to improve its performance in recent years [3, 4].

Differential evolution algorithm (DE) is an optimization algorithm based on the theory of swarm intelligence and solves the optimum problems through competition and cooperation of the population. The operations of DE algorithm include mutation, crossover and selection [5]. DE algorithm is easily carried out, has good local search ability, memories the best solution of all individuals and shares the internal information of the population. However, DE has poor convergence speed and no mechanism to extract and use global information about the search space.

Therefore, this paper proposes a novel hybrid algorithm to solve global optimization problems. It combines the merits of above of two algorithms and imports the chaotic parameter control (CPC) strategy of reference [6] to DE to construct chaotic differential evolution algorithm (cDE) to strengthen DE's exploitation ability. The proposed algorithm has strong exploration and exploitation ability.

2 Sub-optimum Algorithms

2.1 EDA

Let $x(t)$ be the population of solutions at generation t. EDA operates as follows.

Step 1 Selection. Select k promising solutions from the current population $p(t)$ to be excellent population $Q(t)$ by truncation selection. $k = \lfloor truncparam \times N \rfloor$, where *truncparam* is selection rate.

Step 2 Modelling. Build a probability distribution model $P(x)$ based on the statistical information extracted from the solutions in $Q(t)$.

Step 3 Sampling. Sample new solutions according to the constructed probabilistic model $P(x)$.

Step 4 Replacement. Fully or partly replace solutions in $x(t)$ by the sampled new solutions to form a new population $x(t + 1)$.

In this paper, Gaussian model is used in our algorithm. In Gaussian model, the joint density function of the t-th generation is written as follows:

$$P\left(x\right) = \prod_{j=1}^{D} f_{\mathrm{N}}\left(x_j; \mu_j; \sigma_j\right) = \prod_{j=1}^{D} \frac{1}{\sqrt{2\pi\sigma_j}} e^{\frac{1}{2}\left(\frac{x_j - \mu_j}{\sigma_j}\right)^2} \tag{1}$$

In Eq. (1), the D-dimensional joint probability distribution is conducted as a product of D univariate and independent normal distributions. There are two parameters for each variable which include μ_j^k and σ_j^k. They can be estimated as follows:

$$\mu_j^t = \frac{\sum\limits_{i=1}^{k} x_{ij}^t}{k}, \sigma_j^t = \sqrt{\frac{\sum\limits_{i=1}^{k} \left(x_{ij}^t - \mu_j^t\right)^2}{k}}, \tag{2}$$

where $x = (x_{i1}^t, x_{i2}^t, ..., x_{ij}^t, ..., x_{iD}^t)$ is a single individual of t-th generation.

2.2 Chaotic DE

DE is utilized as other sub-optimum algorithm. In order to accelerate the convergence speed and strengthen the search ability of DE, a chaotic strategy named chaotic parameter control (CPC) [6] is utilized. Two mutation operators are selected, which are Eq. (6) [7] and Eq. (7) [7]. Chaotic DE operates as follows.

Step 1 Update the scale factor F, crossover factor CR and differential decisive factor γ with CPC which is display in Eq. (3, 4, 5).

$$F = 4 \times CR \times (1 - CR) \tag{3}$$

$$CR = 4 \times F \times (1 - F) \tag{4}$$

$$\gamma = 4 \times \gamma \times (1 - \gamma) \tag{5}$$

Step 2 If $rand > \frac{\gamma}{2}$, utilizing differential operator 1 of Eq. (6) to construct the mutative individual v_i, or else differential operator 2 of Eq. (7), where $rand$ is a uniformly distributed number in interval [0, 1].

$$v_i = x_{r_1} + F \times (x_{r_2} - x_{r_3}) \tag{6}$$

$$v_i = (F + 0.5) \times x_d + (F - 0.5) \times x_i + F \times (x_b - x_c) \tag{7}$$

Where x_{r_1}, x_{r_2}, $x_{r_3}$are randomly selected from the current population and just $r_1 \neq r_2 \neq r_3 \neq i$. x_d is the best solution of the current population, $(x_b - x_c)$ is a randomly sampled vector differentials with $b \neq c$.

Step 3 Crossover operation with Eq. (8).

$$u_{ij}^{t+1} = \begin{cases} v_{ij}^{t+1}, & if \quad rand < CR \\ x_{ij}^t, & otherwise \end{cases} \tag{8}$$

3 EDcDE Algorithm

With above considerations of two classical optimum algorithms, a new optimum algorithm (EDcDE) is proposed, which combines the merits of both algorithms in order that the population can be preferably evolved.

3.1 The selected factor

Through the selected factor δ, most of individuals in the population are generated by EDA in the initial evolution. EDA forces the search region to quickly approach a promising area as close as possible. In the middle of evolution or later, most of individuals in population are generated by cDE with δ in order to ensure exact search. Thus, δ is adjusted with Eq. (9).

$$\delta_{t+1} = \delta_{\max} - (\delta_{\max} - \delta_t) \times B \tag{9}$$

Where t is the current population, $G_{\max}$ is the maximum value of iteration. $\delta_{\max}$ and $\delta_{\min}$ are the upper and lower bound respectively, B is the rate of change.

3.2 The procedure of EDcDE

Step 1 Initialize the population so that it can evenly distribute in the solution space and parameters, and then set $i = 1$, $t = 1$.

Step 2 Evaluate the target value of the test individual according to the objective function.

Step 3 Compute the mean μ^t and standard deviation σ^t of the current population $x(t)$ with Eq. (2), and then construct the probabilistic model $P(x)$ with Eq. (1). And then compute the selected factor δ with Eq. (9).

Step 4 If $i \leq N$, go to Step 5; otherwise go to Step 6.

Step 5 If $rand < \delta$, the individual η_i^{t+1} is generated by cDE, or else according to the probabilistic model $P(x)$; then, $i = i + 1$, go back to Step 4;

Step 6 Execute the greedy selection operation with Eq. (10), then go to Step 7.

$$x_i^{t+1} = \begin{cases} \eta_i^{t+1}, & if \quad f(\eta_i^{t+1}) < f(x_i^t) \\ x_i^t, & otherwise \end{cases} \tag{10}$$

Step 7 $t = t + 1$, if t attains $G_{\max}$, the EDcDE is terminated; otherwise, go back to Step 2.

4 Convergence Analysis of EDcDE

In this paper, the process of EDcDE is analyzed with Markov processes. First of all, we define $S = R^D$ as the search space of the individual, S^N as population space and $f : S \to R^+$ as the fitness function. Then, the process of proof is displayed as following.

Definition 1 *Let $\{x_t, t \in N^+\}$ denote stochastic processes, if $t \in S$ and random variable sequence $x_1, x_2, ..., x_t \in S$, its condition probability meets: $P\{X_{t+1} = i_{t+1}|X_0 = i_0, X_1 = i_1, ..., X_t = i_t\} = P\{X_{t+1} = i_{t+1}|X_t = i_t\}$. Then $\{X_t, t \in N^+\}$ stochastic process can be described as Markov chain.*

Definition 2 *Let $T_e : S^N \to S$ denote the EDA operator and its probability distribution can be described with Eq. (11) according to Eq. (1).*

$$P(T_e(X) = \eta_i) = \prod_{j=1}^{D} \frac{1}{\sqrt{2\pi\sigma_j}} e^{\frac{1}{2}\left(\frac{x_{ij} - \mu_j}{\sigma j}\right)^2} \tag{11}$$

Definition 3 *Let $T_m : S^N \to S$ denote the mutation operator and its probability distribution can be described according to Eq. (3)-Eq. (7) which can be boiled down with Eq. (12)*

$$\begin{aligned} P(T_m(X) = v_i) = \gamma \times &\sum_{x_{r1}, x_{r2}, x_{r3} \in S^3} P(T_m^1(X) = \{x_{r1}, x_{r2}, x_{r3}, F\}) \\ &+ (1 - \gamma) \times \sum_{x_{r2}, x_{r3} \in S^3} P(T_m^2(X) = \{x_i, x_{best}, x_{r2}, x_{r3}, F\}) \end{aligned} \tag{12}$$

Definition 4 *Let $T_c : S^2 \to S$ denote crossover operator and its probability distribution can be described with Eq. (13), in which k is the number of crossover.*

$$P(T_c(x_i, v_i) = \eta_i) = C_D^k CR^k (1 - CR)^{D-k},\ k = 1, 2, 3, ..., D \tag{13}$$

Definition 5 *Let $T_s : S^2 \to S$denote the selection operator, which selects the better individual between the new individual η_i and original target vectors x_i. And its probability is defined with Eq. (14).*

$$P(T_s(x_i, \eta_i) = x_i(t+1)) = \begin{cases} 1 & f(\eta_i) \le f(x_i) \\ 0 & else \end{cases} \tag{14}$$

Definition 6 *Let $T_p : \delta \to \delta$ denote the selected factor, which decides the number of individuals from DE and EDA respectively, and its probability distribution can be defined with Eq. (15).*

$$P(T_p(\delta) = \delta) = \delta_{\max} - (\delta_{max} - \delta) \times B \tag{15}$$

Theorem 1 *The evolutionary direction of EDcDE is monotonous, that is $f(X(t+1)) \le f(X(t))$.*

Proof According to Eq. (14), the selection operator of EDcDE is greedy selection so as to main the promising individuals of last generation. Therefore, the best fitness of population is monotonous and no ascending.

Theorem 2 *In the EDcDE algorithm, the sequence $\{X(t),\ t \in N^+\}$ of new individual in each generation can be described as the homogeneous Markov chain.*

Proof Due to the sequence of population of EDcDE can be shown as following.

$$X(t+1) = T(X(t)) = T_s \circ ((T_p \circ T_e(X(t)) \oplus ((1 - T_p) \circ (T_c \circ T_m(X(t)))))$$

Where T_s, T_e, T_m, T_p and T_c are not relevant to the time t. And $X(t+1)$ are relevant to $X(t)$. Therefore, $\{X(t), t \in N^+\}$ can be described as Markov chain, whose transition probability is presented as following.

$$\begin{aligned} P(T(X(t))_i = x_i(t+1)) = \sum_{\eta_i \in S} & P(T_s(X(t), \eta_i) = x_i(t+1)) \\ & \bullet \{(P(T_p(\delta) = \delta) \bullet P(T_e(X(t)) = \eta_i) + (1 - P(T_p(\delta) = \delta)) \\ & \bullet \sum_{v_i \in S} P(T_c(X(t), v_i) = \eta_i) \bullet P(T_m(X(t)) = v_i)\} \end{aligned}$$

And due to $\forall X(n) \in S^N$, $\exists\{x_{r1}, x_{r2}, x_{r3}\} \in S^N$ and $v_i, \eta_i \in S^N$, there exits $P(T_m^1(X) = \{x_{r1}, x_{r2}, x_{r3}, F\}) > 0$, $P(T_m^2(X) = \{x_i, x_{best}, x_{r2}, x_{r3}, F\}) > 0$, $P(T_s(x_i, \eta_i) = x_i(t+1)) > 0$, $P(T_e(X(t)) = \eta_i) > 0$, $P(T_c(X(t), v_i) = \eta_i) > 0$, $P(T_p(\delta) = \delta) > 0$, $\gamma = 4 \times \gamma \times (1 - \gamma) > 0$. Therefore, $P(T(X(t))_i = x_i(t+1))$ which is not relevant to time t and its transition probability can be described with $P(T(X(t)) = X(t+1)) = \prod_{i=1}^{N} P(T(X(t))_i = x_i(t+1)) > 0$. The sequence $\{X(t), t \in N^+\}$ can be described as homogeneous, irreducible, and nonperiodic Markov chain.

Theorem 3 *The sequence $\{X(t),\ t \in N^+\}$ of EDcDE can converge to the satisfied population set M^* with the probability of 1, that is for which is original state, there exists Eq. (16).*

$$\lim_{t\to\infty} P\{X(t) \in M^*|X(0) \in X_0\} = 1 \tag{16}$$

Proof According to the problem definition, we can define $F(X(t)) = \min(f(X(t)_i),\ i = 1, 2, 3...,$ n, and according to theorem 2, the state-transition matrix $P_t\{A,\ B\}$ and the transition probability $P(t)$ are displayed as follows:

$$P_t\{A, B\} = P\{X(t+1) = B|X(t) = A\}, P(t) = (P_t\{A, B\}; A, B \in S^N)$$

According to definition 5 and theorem 1, there exists $p_s^t = \begin{cases} 0 \ , & F(A) < F(B) \\ 1 \ , & F(A) \geq F(B) \end{cases}$. Thus, if $F(B) \leq F(A)$, then $\lim\limits_{t\to\infty} P_t\{A, B\} = P(T_s \circ ((T_p \circ T_e(X(t)) \oplus ((1 - T_p) \circ (T_c \circ T_m(X(t)))))) > 0$ else if $F(B) > F(A)$, then $\lim\limits_{t\to\infty} P_t\{A, B\}$. Let $P(\infty) = \lim\limits_{t\to\infty} P_t(t) = P_\infty\{A, B\}, A, B \in S^N$. So we have $P_\infty(A, B) = \begin{cases} = 0 \ , & F(A) < F(B) \\ > 0 \ , & F(A) \geq F(B) \end{cases}$. Obviously, $P(\infty)$ is a random matrix. Since M^* is a nonperiodic positive recurrent class and $\overline{M} = S^N - M^*$ is a nonrecurrent class, $\{X(t), t \in N^+\}$ has strong ergodicity. As for $\forall X_0$, there exists $\lim_{t\to\infty} P\{X(t) \in B|X(0) = X_0\} = \pi_\infty(B)$ according to theorem 2, where $\sum_{B\in M^*} \pi_\infty(B) = 1$. Therefore, $\lim\limits_{t\to\infty} P\{X(t) \in M^*|X(0) \in X_0\} = \sum_{B\in M^*} \pi_\infty(B) = 1$.

5 Experiments and Results

5.1 Test functions and experimental setup

We use 6 benchmark test functions to test the performance of the proposed algorithm in Table 1. EDcDE is compared with the PBILc [8] and the conventional DE [9]. The learning rate of PBILc is set to 0.2 and the selection rate to 0.1. As for DE, the parameters setting suggested by the original work [9] is employed: $F = 0.5$, $CR = 0.6$. The parameters of EDcDE are set as follows: $CR = 0.3$, γ, *truncparam* and $\delta_{\min}$ are all set to 0.2, $\delta_{\max} = 0.9$, $B = 0.99$. Meanwhile, we set $D = 30$, $N = 150$ and $G_{\max} = 2000$ for all algorithms. Each experiment is terminated either when the current iteration attains the maximum iteration. All results are averaged over 20 runs.

Table 1: Benchmark test functions

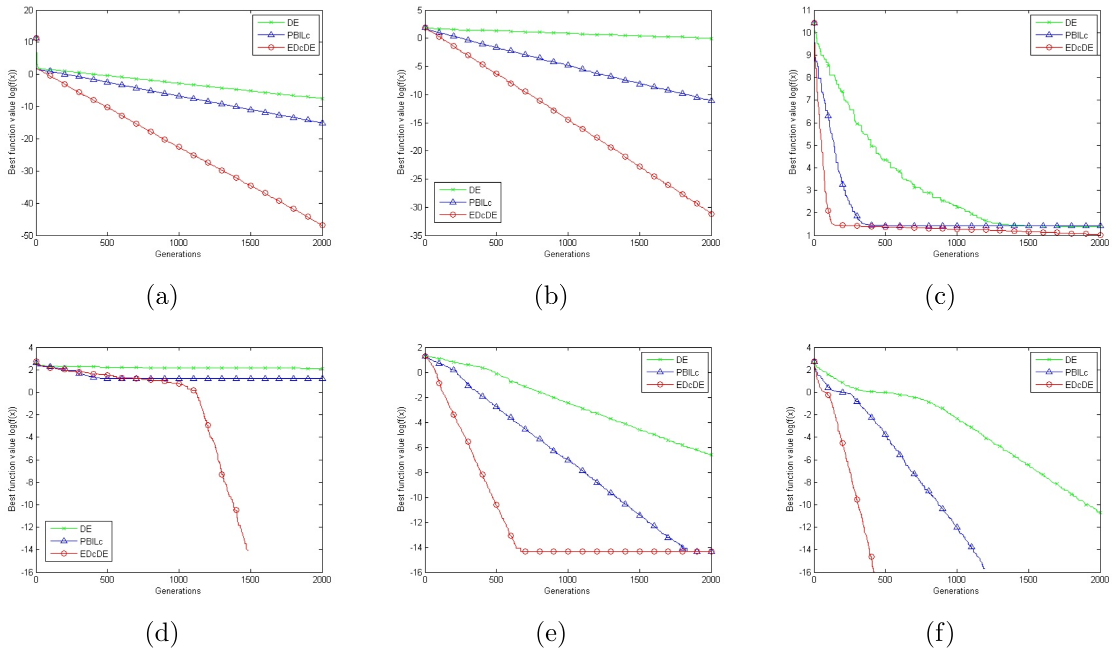
Function expression	Value range
$f_1(x) = \sum_{i=1}^n \lvert x_i\rvert + \prod_{i=1}^n \lvert x_i\rvert$	$[-10, 10]^n$
$f_2(x) = \max\limits_{1\leq i\leq n} \lvert x_i\rvert$	$[-100, 100]^n$
$f_3(x) = \sum_{i=1}^n (100({x_i}^2 - x_{i+1})^2 + (x_i - 1)^2)$	$[-100, 100]^n$
$f_4(x) = \sum_{i=1}^n ({x_i}^2) - 10\cos(2\pi x_i) + 10$	$[-5.12, 5.12]^n$
$f_5(x) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n {x_i}^2}) + e - exp(\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)) + 20$	$[-32, 32]^n$
$f_6(x) = \frac{1}{4000}\sum_{i=1}^n {x_i}^2 + 1 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}})$	$[-600, 600]^n$

Table 2: Experimental results

	Algorithms	Mean	Best		Algorithms	Mean	Best
f_1	EDcDE	4.67e-47	5.75e-48	f_4	EDcDE	0.00e+00	0.00e+00
	PBILc	4.67e-47	5.75e-48		PBILc	1.32e+01	7.96e+00
	DE	3.57e-08	2.60e-08		DE	1.23e+02	1.02e+02
f_2	EDcDE	7.92e-32	1.09e-32	f_5	EDcDE	4.44e-15	4.44e-15
	PBILc	9.46e-04	3.88e-12		PBILc	4.44e-15	4.44e-15
	DE	1.12e+00	9.40e-01		DE	2.21e-07	1.69e-07
f_3	EDcDE	1.13e+01	1.08e+01	f_6	EDcDE	0.00e+00	0.00e+00
	PBILc	9.40e+01	2.59e+01		PBILc	3.70e-04	0.00e+00
	DE	2.41e+01	2.38e+01		DE	6.09e-11	2.28e-12

5.2 Experimental results

The statistical results are summarized in Table 2 and the evolution evolutionary curves of all functions are shown in Fig. 1. As seen form the Table 2 and Fig. 1, for all functions, EDcDE is the clear winner in term of overall performance which includes result accuracy and convergence speed. As the strong disturbance of its chaotic sequence and the diversity of population maintained by adopting two efficient differential mutation operators, EDcDE have good exploitation especially for complex global optimization. In addition, EDA plays an important role for the initial exploration. Therefore, EDcDE have excellent performance for global optimization.



(a) (b) (c)

(d) (e) (f)

Fig. 1: Evolutionary curves of function f_1-f_6, (a)-(f)

6 Conclusion

In this paper, we propose a new hybrid algorithm which called EDcDE. EDcDE makes one part of a trial individuals generate from the DE mutation and crossover and the other part of the trial individuals are from EDA through the selected factor. The local information and the global information are effectively incorporated together. The local search ability of DE and the global search ability of EDA are fully exploited. Experimental evidence is provided that the proposed algorithm is better than DE and PBILc. Therefore, EDcDE has excellent performance for global optimization problem.

Acknowledgement

This work was financially supported by the National Natural Science Foundation of China under grant numbers 51365030. It was also supported by the General and Special Program of the Postdoctoral Science Foundation of China, the Science Foundation for Distinguished Youth Scholars of Lanzhou University of Technology, Lanzhou Science Bureau project under grant numbers 2012M521802, 2013T60889, J201405, and 2013-4-64, respectively.

References

[1] P. Larranga, A. J. Lozano, Estimation of distribution algorithms: A new tool for evolutionary computation, Springer, New York, 2002.

[2] S. H. Chen, M. C. Chen, P. C. Chang, Guidelines for developing effective estimation of distribution algorithms in solving single machine scheduling problems, Expert Systems with Applications, 37 (32), 2010, pp. 6441-6451.

[3] L. Wang, C. Fang, A hybrid estimation of distribution algorithm for solving the resource-constrained project scheduling problem, Expert Systems with Applications, 39 (3), 2012, pp. 2451-2460.

[4] C. E. Izquierdo, J. L. Gonzalez Velarde, B. M. Batista, J. M. Moreno-Vega. Hybrid Estimation of Distribution Algorithm for the Quay Crane Scheduling Problem. Applied Soft Computing, 13 (10), 2013, pp. 4063-4076.

[5] H. Yan, R. Li, A novel discretization algorithm based on differential evolution in rough sets, Journal of Computational Information Systems, 8 (5), 2012, pp. 1937-1944.

[6] Y. Wang, B. Li, X. Lai, Variance priority based cooperative co-evolution differential evolution for large scale global optimization, in: Proc. Evolutionary Computation, '09, 2009, pp. 1232-1239.

[7] S. Das, P. N. Suganthan. Differential evolution: A survey of the state-of-the-art. Evolutionary Computation, IEEE Transactions on, 15 (1), 2011, pp. 4-31.

[8] M. Sebag, A. Ducoulombier. Extending population-based incremental learning to continuous search spaces, in: Proc. the 5th International Conference on Parallel Problem Solving from Nature, 1998, pp. 418-427.

[9] R. He, Z. Yang. Differential Evolution with Adaptive Mutation and Parameter Control Using Levy Probability Distribution, Journal of Computer Science and Technology, 27 (5), 2012, pp. 1035-1055.