TELKOMNIKA, Vol.11, No.1, January 2013, pp. 123~129
ISSN: 2302-4046

Estimation of Distribution Immune Genetic Algorithm and Convergence Analysis

LIU Zhen*[1], HU Yun-an[1], SHI Jian-guo[2]
[1]Dept. of Control Engineering, Naval Aeronautical and Astronautical University, Yantai, China
[2] Dept. of No.7, Naval Aeronautical and Astronautical University, Yantai, China
*corresponding author, e-mail: hylz1008@126.com, hya5071@sina.com, cherrapple@126.com

Abstract

In the traditional immune genetic algorithm, crossover and mutation can disrupt the superior chromosome, so make the algorithm took a long time to converge to the best solution. The way of crossover and mutation based on marginal product model which can make the algorithm converge quickly was proposed in order to avoid the disruption of the superior chromosome. The pseudo parallel evolution mechanism was also brought into the proposed algorithm in order to enhance the diversity of the population. The convergence character of the algorithm is analyzed. The model theorem of estimation of distribution immune genetic algorithm was given and the convergence rule was also given. Simulation results of several benchmark functions show that the proposed algorithm is superior than genetic algorithm immune genetic algorithm. So the proposed algorithm is correct and feasible.

Keywords*: immune genetic algorithm, estimation of distribution, marginal product model, ectended compact*

1. Introduction

Artificial immune system is a computational intelligence paradigm inspired by the biological immune system, and has been applied successfully to a variety of optimization problems. Immune genetic algorithm (IGA) is a novel bio-immune evolutionary algorithm, which have gained more and more attention due to their excellent adaptation and robustness. In the past several years, some people have proposed many kinds of improved artificial immune algorithms.

In the widely studied immune algorithms, many results are about the clonal selection, Wu proposes an adaptive quantum immune clonal algorithm and the convergence of the proposed algorithm is proved theoretically [1]. Liu introduces the biologic information mechanism into the immune algorithm and the biologic information mechanism is employed to improve information interaction and accelerate the speed of evolution [2]. A novel adaptive immune algorithm, which is based on immune danger theory, was proposed by Xu to emulate the entire immune mechanisms and to enhance the performance for complex multimodal problems [3]. Chen develop a hybrid immune multi-objective optimization algorithm based on clonal selection principle, a hybrid mutation operator is proposed with the combination of Gaussian and polynomial mutations [4]. Ali presents a new hybrid optimization approach based on immune algorithm and hill climbing local search algorithm in order to enhance the running efficiency [5].

IGA has been proposed and is characterized by rapid convergence and high convergence precision. Evolution parameters and evolution approach should be selected properly so as to contribute positively to the optimization performance of IGA. However, in most of the existing IGAs, the crossover and mutation always disrupt the superior pattern of chromosome, then cannot guarantee the global optimum. So in order to promote the evolutionary performance of the IGA, exploration and exploitation can be provided simultaneously when the crossover and mutation can be set correct.

Estimation of distribution algorithms (EDA) have been proposed as an extension of genetic algorithms. EDAs uses probability distributions derived from the function to be optimized

Received September 28, 2012; Revised November 22, 2012; Accepted November 30, 2012

to generate search points instead of crossover and mutation as done by genetic algorithms. In both cases a population of points is used and points with goodness are selected either to estimate a search distribution or to be used for crossover and mutation. The extended compact genetic algorithm (ECGA) is a class of EDAs which is proposed by Harik [6]. The algorithms replaces traditional variation operators of genetic and evolutionary algorithms by building a probabilistic model of promising solutions and sampling the model to generate new candidate solutions. The ECGA uses the marginal product model (MPM), which can be divided into two components, (1) a partition over the variables, defining which variables are independent and which variables are linked, and (2) a probability distribution over each partition. So in the IGA, the chromosome can be divided into several MPMs and the superior individuals can be reserve after iterations. Tan introduces the quantum-inspired evolutionary algorithm into EDAs, and proposed a new quantum-inspired estimation of distribution algorithm [7], in the paper, the basic idea of EDAS is brought into IGA, and estimation of distribution immune genetic algorithm (EDIGA) is proposed.

The remainder of the paper is organized as follows. The basic idea of EDIGA is introduced in section 2, while section 3 gives the convergence proof of the proposed algorithm. Section 4 presents the simulations results for the benchmark functions. Lastly, section 5 draws the conclusions.

2. Estimation of Distribution Immune Genetic Agorithm

2.1. Population Initialization

Population initialization is a crucial task in evolutionary algorithms because it can affect the convergence speed and also the quality of the final solution. If no information about the solution is available, then random initialization is the most commonly used method to generate candidate solutions (initial population). But random initialization can lead population to local extremum, so in order to enhance the diversity of the population, and accelerate the convergence speed, the paper adopts the opposition-based learning to generate initial population [8].

2.2. Crossover and Mutation Based on MPM

The traditional crossover and mutation always lead the algorithm trap into local optimum, and the better pattern in the chromosome can be break through crossover and mutation, the key that the evolutionary algorithm can converge is that the better pattern can grow increasing in the next iterations. So the better pattern should be reserve. So in the paper, the crossover and mutation based on MPM is proposed.

Definition1 The allele in the chromosome can be divided into several MPMs, and every MPMs can be viewed as a whole. The crossover can perform according to the MPMs, the crossover is named as MPM crossover.

The MPM crossover can guarantee the better individuals grow increasing and the linkage between the allele can not break in the process of the MPM crossover. The detailed procedure can be descried as: randomly select two individuals from the population and the two individuals can exchange the selected MPM through the crossover operator.

As shown in the Fig1, the chromosome is q_i and q_j respectively, and the different shape represents the different MPMs, the gene 1, 3 and 4 form the MPM1, the gene 2 forms the MPM2, the allele 5 and 6 can be viewed as MPM3. When we choose the MPM1 randomly, the crossover process can be shown as Figure 1.

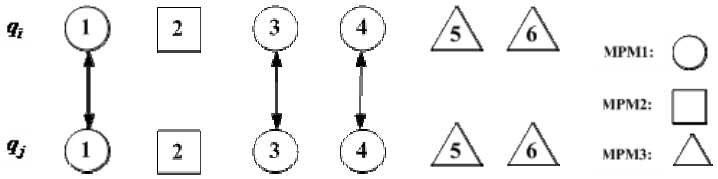


Figure 1. Sketch map of MPM crossover

The population evolves the optimum slowly and can often meet the local extreme when the allele that chooses to mutated randomly, so the paper proposed a new kind of mutation based on MPM.

Definition2 Choose a MPM from the population individuals, and the allele in the chosen MPM can be mutated according to the mutation probability p_m.

In the running process of the evolutionary algorithm, if the evolutionary algorithm always adopts the same mutation probability, the convergence of the EA will slow and the better individuals cannot always reserve in the next generation, so if the mutation probability can change adaptively according to the evolution condition, so the mutation probability of the ith individuals can be set as:

$$p_m = p'_m \frac{f_{\max}(t) - f_i(t)}{f_{\max}(t) - f_{\min}(t)} \tag{1}$$

In the equation (1), $f_{\max}(t)$ and $f_{\min}(t)$ is the maximum and minimum fitness at the iteration t, and p'_m is the predefined constant.

2.3. Immune Selection

In order to promote the diversity of the evolution population, and make the selection mechanism direct the correct search, the fitness sharing is brought into the EDIGA. Fitness sharing mechanism can avoid the local convergence and high selection pressure. For the two antibody $a_i(t)$ and $a_j(t)$, the sharing function can be defined as:

$$Sh(d_{ij}) = \begin{cases} 1 - \left(\dfrac{d_{ij}}{\sigma_s}\right)^2, & d_{ij} < \sigma_s \\ 0, & otherwise \end{cases} \tag{2}$$

In the equation (2), σ_s is the sharing radius, d_{ij} is the distance between the ith and the jth. When the problem scale is n, and the individuals $a_i(t)$ can be defined as:

$$f_i(t) = f_i^*(t) \Big/ \sum_{j=1}^{n} Sh(d_{ij}) \tag{3}$$

$f_i^*(t)$ and $f_i(t)$ are the fitness before and after fitness sharing mechanism. In order to make sure that the population can keep diversity, the selection probability for the ith individual can be set as [9]:

$$p_i(t) = \mu \frac{f_i(t)}{\sum_{j=1}^{N} f_j(t)} + (1-\mu)\frac{1}{N} e^{(-C_i(t)/v)} \tag{4}$$

In the equation (4), μ and v are the adjustive parameter and are all constant, they can be set as 0.8 and 1.2 respectively, $C_i(t) = \dfrac{h_i(t)}{n}$, $h_i(t)$ represents the number of individuals whose fitness is between $[f_i(t), f_i(t) + \Delta f(t)]$, $\Delta f(t) = (f_{\max}(t) - f_{\min}(t))/3$.

2.4. The Individuals Migration Strategy

In order to promote the diversity of the population, the whole population can be divided into two subpopulation independently [10], the two subpopulation can exchange individuals

between them.

$$\eta = \begin{cases} \max\left\{1, \dfrac{|\Delta fit|}{\max\{fit_i, fit_j\}}\right\} & |\Delta fit| > \theta \\ 0 & |\Delta fit| \le \theta \end{cases} \quad (5)$$

In the equation (5), $\Delta fit = fit_i - fit_j$, fit_i and fit_j are the best fitness of the two subpopulation. In every iteration, the migration method can be determined as follows:

(1) if $\Delta fit > 0$, then the individuals can migrate from the ith subpopulation to the jth subpopulation, else the individuals will migrate to the ith subpopulation, if $\Delta fit = 0$, do nothing;

(2) if $\eta > 0$, then the scale of the migration is λn, n is the population scale.

2.5. The Process of the Proposed EDIGA

Above all, after the detailed description of the proposed algorithm, the process of the proposed algorithm can be summarized as:

Step1 Initialize the parameter of the algorithms, such as the problem scale n, the chromosome length m et al, then Initialize chromosome according to section 2.1;

Step 2 Perform the MPM crossover and the MPM mutation operation, crossover probabltiy can be set as 0.7 and the mutation probability can be set according to equation (1);

Step 3 Calculate the appetency between the antibody and antigen, the fitness can got according to equation (3), and the selection probability can be got according to equation (4);

Step 4 If the termination criterion is satisfied, this procedure ends; otherwise, set t = t + 1, and return to Step 2.

3. Convergence of the Proposed Algorithm

Theorem 1 In the EDIGA, the proposed algorithm adopts crossover and mutation based on MPM, then model defined distance can not affect the expectation of the pattern in the next iteration. The low order model individuals whose fitness are better than the average population fitness will grow exponentially.

Proof The proposed algorithm adopts the MPM crossover and MPM mutation. Because MPM crossover and mutation can not disrupt the pattern. Suppose that the model number of the t th generation is $m(\zeta, n)$, and the mutation probability is p_m, then the probability that all the allele do not mutated is $(1-P_m)^{O(\zeta)}$, in the most case $P_m \ll 1$, and $(1-P_m)^{O(\zeta)} = P_m \bullet O(\zeta)$, $O(\zeta)$ is the model defined order. Then according to the model theorem, the model survival probability is:

$$m(\zeta, n+1) \ge m(\zeta, n)\frac{\overline{f}(\zeta)}{\overline{f}_n}[1 - p_{va} \bullet O(\zeta) - p_m \bullet O(\zeta)] \quad (9)$$

In the equation (9), $\overline{f}(\zeta)$ is the average fitness of $(\zeta \cap \vec{X}(n))$, and $\overline{f}_n$ is the average fitness of the whole population. If the fitness of the model is k times of the average fitness, then $\overline{f}(\zeta) = (1+k)\overline{f}_n$, and we can get:

$$m(\zeta, n+1) \ge m(\zeta, 0)(1+k)^n[1 - p_{va} \bullet O(\zeta) - p_m \bullet O(\zeta)] \quad (10)$$

Theorem 2 The population of the proposed EDIGA $\overline{X}(t)$ is a finite Markov process.

Proof Because the state space of the proposed EDIGA is finite, suppose the encoding lenghtn of chromosome is χ, the individuals can be change from $(0,0,,...,0)$ to $(\chi-1, \chi-1, ..., \chi-1)$. Then the whole state space is $\Omega = \chi^{NL}$, and N is the number of subpopulation, L is the lenghtn of chromosome. Then the state space can be constructed

through choosing N individuals form Ω, the state space is finite. Also because $\overrightarrow{X}(t+1)=T_V \circ T_G(\overrightarrow{X}(t))$, T_V and T_G have no relation with the iteration number, T_V is the migration operator, T_G is the genetic operator, and $T_G = T_M \circ T_C \circ T_S$, T_M is the mutation operator, T_C is the crossover operator, T_S is the immune selection operator. So the $\overrightarrow{X}(t+1)$ has no relation with $\overrightarrow{X}(t)$, then $\overrightarrow{X}(t)$ is the finite Markov process.

Lemma 1 [11] set $\overline{a}_t^B = P(\overrightarrow{X}(t+1) \bigcap B^c \neq \varnothing / \overrightarrow{X}(t) \bigcap B^c = \varnothing)$,

$\overline{\beta}_t^B = P(\overrightarrow{X}(t+1) \bigcap B^c \neq \varnothing / \overrightarrow{X}(t) \bigcap B^c \neq \varnothing \phi)$, if $\overline{a}_t^B$, $\overline{\beta}_t^B$ satisfy (1) $\sum_{t=1}^{\infty}(1-\overline{\beta}_t^B) = \infty$, (2) $\frac{\overline{a}_t^B}{1-\overline{\beta}_t^B} = 0$, then $\{\overrightarrow{X}(t)\}$ can probability strong convergence to the satisfaction set B, then $\lim_{t\to\infty} P(\overrightarrow{X}(t) \subset B) = 1$。

Theorem 3 The evolution population of the proposed EDIGA is $\overrightarrow{X}(t)$, then $\overrightarrow{X}(t)$ is probability strong convergence to the satisfaction set B.

Proof From the evolution process of the EDIGA, when the population contains the satisfaction set in the tth iteration, then the population will also contains the satisfaction set in the next generation, then $P(\overrightarrow{X}(t+1) \bigcap B^c \neq \varnothing / \overrightarrow{X}(t) \bigcap B^c = \varnothing) = 0$, so $\overline{a}_t^B = 0$, and the condition (2) can meets. Condition (1) is the same as $\lambda = \sup(\overline{\beta}_t^B) < 1$.

Because $P(\overrightarrow{X}(t+1) \bigcap B^c \neq \varnothing / \overrightarrow{X}(t) \bigcap B^c \neq \varnothing) + P(\overrightarrow{X}(t+1) \bigcap B^c = \varnothing / \overrightarrow{X}(t) \bigcap B^c \neq \varnothing) = 1$,

then we can get $\overline{\beta}_t^B = 1 - P(\overrightarrow{X}(t+1) \bigcap B^c = \varnothing / \overrightarrow{X}(t) \bigcap B^c \neq \varnothing)$,

$P(\overrightarrow{X}(t+1) \bigcap B^c = \varnothing / \overrightarrow{X}(t) \bigcap B^c \neq \varnothing) > 0$. It is proves that $P(\overrightarrow{X}(t) \subset B)$ is least than 1 in no more than one condition, then at the generation $t+1$, $P(\overrightarrow{X}(t+1) \subset B) = 1$ so $\lambda = \sup(\overline{\beta}_t^B) < 1$, and the condition can also meets, then $\overrightarrow{X}(t)$ can probability strong convergence to the satisfaction set.

4. Simulation Results of Benchmark Functions

In order to evaluate the algorithm, the proposed algorithm has been applied to the optimization of well-known benchmark functions and its performance is compared with that of other algorithm. The dimension of f_1 and f_2 can be set as 2, and the dimension of f_3 and f_4 can be set as 20, the optimum of function f_1 is -1.1031628, and the optimum of f_2, f_3 and f_4 are all zero.

$$f_1(x)=(4-2.1x_1^2+\frac{x_1^4}{3})x_1^2+x_1x_2+(-4+4x_1^2)x_2^2, x_i \in [-3,3],$$

$$f_2(x)=100(x_2-x_1^2)^2+(x_1-1)^2, x_i \in [-2.048,2.048],$$

$$f_3(x)=-20\left(-0.02\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e,\ \ x_i \in [-30,30],\ \ n=20,$$

$$f_4(x)=1+\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right), x_i \in [-600,600],\ \ n=20.$$

The scale of population scale is 100 and $p_c = 0.7$, $p_m = 0.05$. The maximal iteration can be set as 100. The proposed algorithm can be compared with genetic algorithm (GA) and immune genetic algorithm (IGA).

Table 2 is the statistical results compared with GA and IGA. From the simulation results, the convergence speed of the proposed algorithm is faster than GA and IGA, and can also get

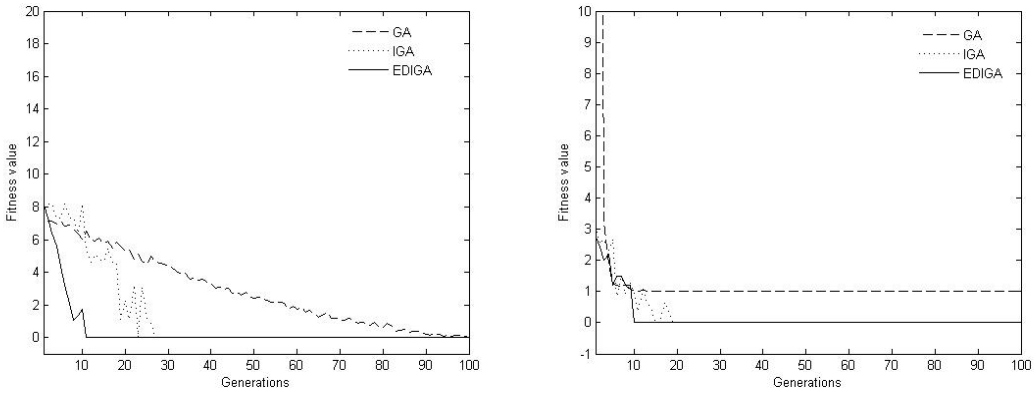
better results. From the statistical results, the proposed algorithm can get better results, and the simulation results is also stable.

Table 1. Simulation results of the three algorithms

Function	GA		IGA		EDIGA	
	Mean	Std	Mean	Std	Mean	Std
f_1	-1.02	3.25×10^{-2}	-1.03	2.85×10^{-2}	-1.03	2.89×10^{-2}
f_2	7.01×10^{-1}	4.19×10^{-1}	6.67×10^{-2}	4.08×10^{-2}	6.10×10^{-2}	3.65×10^{-2}
f_3	1.03	8.01×10^{-1}	2.18×10^{-2}	1.64×10^{-2}	4.31×10^{-3}	2.10×10^{-3}
f_4	6.54×10^{-1}	3.56×10^{-2}	9.42×10^{-2}	2.17×10^{-2}	$5.17.\times10^{-3}$	2.31×10^{-3}

To the function f_1 and f_2, the simulation results of EDIGA is only a little better than the simulation results of GA and IGA. It is because that the MPM in the low-order function contain less allele, sometimes the MPM only contains one allele, so it has small effect on the convergence of the proposed evolution algorithm. But to the high-order function f_3 and f_4, the MPM can enhance the performance of the proposed algorithm obviously.

The iteration running results of function f_2 and f_3 are shown as Fig2(a) and Fig2(b). From the simulation results, we can see that the proposed algorithm perfoem well than GA and IGA. To the low-order function f_2, each of three algorithms can reach the global optimum, but the proposed algorithm have faster convergence speed, for the high-order function f_3, EDIGA can get better results than the other two algorithms.



(a) simulation results of f_2 (b) simulation results of f_3

Figure 2. Comparison of simulation results

5. Conclusion

The objective of this work is to creat an efficient and competent EA, capable of solving large scale difficult problems. Particularly, we are interested in creating an algorithm that can deal efficiently with problem sub-structures, solving a broader class of problems than the class the IGA can handle. The individuals in the population can crossover and mutation based on MPM and every subpopulation in the proposed algorithms can not only evolve independently, but also exchange information with each other. Simulation results of benchmark functions prove the correctness of the proposed algorithm.

References

[1] Wu QY, Jiao LC, Li YY, et al. Self Adaptive Quantum-inspired Immune Clone Algorithm and Its Convergence Analysis. *Pattern Recognition and Artificial Intelligence*. 2008; 21(5): 592-597.

[2] Liu SR, Zhang BT. Quantum Immune Clonal Algorithm Based on Biologic Information Mechanism. *Pattern Recognition and Artificial Intelligence*, 2008; 21(5): 592–597.
[3] Xu B, Zhuang Y. Adaptive Immune Algorithm Based on Danger Theory. *Journal of Si Chuan University (Engineering Science Edition*. 2011; 43(3): 123–132.
[4] Chen JY, Lin QZ, Ji Z. A Hybrid Immune Multi-objective Optimization Algorithm European. *Journal of Operational Research*. 2010; 204: 294–302.
[5] Ali RY. *A Novel Hybrid Immune Algorithm for Global Optimization in Design and Manufacturing. Robotics and Computer-Integrated Manufacturing*. 2009; 25: 261–270.
[6] GR Harik, FG Lobo, DE Goldberg. The Compact Genetic Algorithm. *IEEE Transactions on Evolutionary Computation*. 1999; 3(4): 287-297.
[7] Tan LX, Guo L. *Quantum-Inspired Estimation of Distribution Algorithm Based on Comprehensive Learning. Pattern Recognition and Artificial Intelligence*. 2010, 23(3): 314-319.
[8] Shahryar R, Hamid RT, Magdy MS. A Novel Population Initialization Method for Accelerating Evolutionary Algorithms. *Computers and Mathematics with Applications*. 2007, 53(10): 1605–1614.
[9] Cheng LJ, Ding YS, Hao KR. *An Ensemble Kernel Classifier with Immune Clonal Selection Algorithm for Automatic Discriminant of Primary Open-angle Glaucom*, Neurocomputing, 2012, 83(15): 1-11.
[10] Zhao FY, Ma Zhen-yue, Wang Yi-bo. Identification of Dynamic Parameters Based on Adaptive Pseudo-parallel Genetic Algorithms. *Journal of Dalian University of Technology*. 2007; 47(2): 252-256.
[11] Zhang WX, Liang Y. *The foundation of genetic algorithm*. Xian: Xi an Jiaotong Publisher, 2003.