Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2015, 7(10):30-38

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

An improved hybrid immune algorithm for multimodal optimization

Jia Liyuan, Zhou Cuihong, He Jianxin and Tu Li

School of Information Science and Engineering, Hunan City University, Yiyang Hunan China

---

ABSTRACT

To improve the efficiency of basic artificial immune algorithm(AIA),this paper presents an improved hybrid immune algorithm(IHIA) for constrained optimization .To maintain high diversity ,the fitness of each individual and concentration were taken into account in determining reproduction probability. Many benchmark functions were used to demonstrate the validity of IHIA and the role of each design of IHIA. Numerical experiments show that IHIA can reduce the complexity of comutation, and then increases the efficiency of IHIA with the maintenance of diversity and convergence in optimizing constrained functions.

Keywords: affiity, information entropy, Guotao algorithm, adaptive mutation, constrained function

---

INTRODUCTION

The rapid development of computer science and technology have changed human life greatly, when people get huge convenience, they also had to face the increasing scale of engineering optimization problems, and the complexity of optimization problems is also increasing, a lot of problems are Non-deterministic Polynomial, these optimization problems can be transformed into function optimization problem through mathematical modeling, so it is very important.to study function optimization problem.

The immune system has distribution, adaptability, diversity of population, self-organization, fast response and other characteristics, artificial immune system can solve a large number of nonlinear problems. To solve engineering problems with immune algorithm also caused great concern of scholars.

Artificial immune algorithm simulates the natural immune system reproductive strategy based on the concentration of antibody,it suppresses a high concentration of the solution,and it promotes low concentration solution,so it maintains the diversity of results,and it prevents the convergence of the system before maturity converge.

EXPERIMENTAL SECTION

2. Improved Hybrid Immune Algorithm(IHIA)

2.1 Immune Algorithmn

Immune algorithm is a new biological intelligence algorithm based on human immunology. It simulates the antigen processing of biological immune system Including the production of antibodies, autologous, clonal expansion, immune memory. These are the main definitions.

2.1.1 Affinity

Definition 1:

If there are M antibodies, there are N genes in each antibody, there are S symbols to be choosed from each gene, they are $k_1, k_2k_3....k_s$,so the information entropy of the M antibodies is shown in formula (1):

$$H(M) = \frac{1}{N}\sum_{i=1}^{M} H_i(M) \tag{1}$$

where， $H_i(M) = \sum_{j=1}^{s} - p_{ij} \log p_{ij}$ ， $H_i(M)$ is the information entropy of the i-th bit of M antibodies.

Definition 2:
Affinity is used to describe the degree of similarity between two antibodies.

The affinity between the v-th antibody and the w-th antibody is defined as formula (2):

$$a_{vw} = \frac{1}{1+H(2)} \tag{2}$$

where， $H(2)$ is the information entropy between the v-th affinity, if $H(2)$ is equal to 0,this means that all genes of the two antibodies are the same, the value of parameter a_{vw} is between 0 and 1.

In a similar way, the affinity between antibody and antigen is defined as $a_v = \frac{1}{1+d_v}$.

Where parameter d_v is the association degree between antibody and antigen, it is used solution space to measure. the value of parameter a_v is between 0 and 1.When d_v is equal to 0, the value of parameter a_v is equal to 1, this means the antibody matches the antigen very well, It is the optimal solution

2.1.2 Concentration of the antibody
Definition 2:
Concentration of the antibody is used to describe the scale among antibodies. The concentration of the v-th antibody is defined as formula (3):

$$c_v = \frac{1}{M\sum_{W=1}^{N} b_{vw}} \quad , \tag{3}$$

where $b_{vw} = \begin{cases} 1, & \lambda a_v \le a_w \le \delta a_v \\ 0, & other \end{cases}$ ， λ, δ are adjustment factors，parameter λ is slightly smaller than 1,and parameter δ is slightly bigger than 1.

2.1.3 Step of general immune algorithm
Step 1:Define the antigen
Convert the problem to be solved into the form of the antigen that matches immune system, and the antigen recognition is the solution of the problem;

Step 2: Define the initial antibody population
Define the population as the problem of antibody solution, evaluate the affinity between antibodies and antigens: The higher the affinity, the better solution of the equation is.

Step 3:Calculatie the affinity between antibodies and antigens.

Step 4: Clonal Selection
Have the greater affinity of antibodies breed first, inhibit the high concentration of antibodies and Eliminate the low concentration of antibodies.

Step 5: Evaluate the new antibody group
If it does not meet the termination condition, then turn to step (3), restart the calculation, if it meets the termination

condition, then the current antibody is the optimal solution.

2.2 Improved Hybrid Immune Algorithmn(IHIA)
2.2.1 Clonal selection operator

The clonal selection operator is the result by considering the antibodies and concentration of antibodies.It is defined as formula (4).

$$p_{exc}(i) = \frac{[f(i)]^{\alpha} \cdot [c(i)]^{-\beta}}{\sum_{i=1}^{M} [f(i)]^{\alpha} \cdot [c(i)^{-\beta}]} \tag{4}$$

Where $p_{exc}(i)$ is the clonal selection probability of antibodies, $f(i)$ is the affinity of the i-th antibody, $c(i)$ is the concentration of the i-th antibody, α, β are the inspired factor of affinity and the inspired factor of concentration.

2.2.2 Selection operator of variable threshold
The selection operator can be determined based on the expected probability of antibodies. The selection operator of variable threshold is defined as formula (5).

$$T(i) = \begin{cases} 1, & p_{exc}(i) \ge T \\ 0, & p_{exc}(i) \le T \end{cases} \tag{5}$$

Where $p_{exc}(i)$ is the expected probability of the i-th unit, $T(i)$ is the actual probability of the i-th unit, T is the threshold[10], in a standard immune algorithm the threshold T is fixed. In our improved hybrid immune algorithmn(IHIA), in order to accelerate the convergence rate, we adjusted the value T. In the early evolution, the threshold value is small, in this case, great affinity individuals and small antibodies concentrations of individuals were selected , the clonal opportunity increased greatly; When the algorithm evolved to a global convergence, the threshold value is large, in this case, the clones opportunity of great affinity individuals and small antibodies concentrations of individuals reduced greatly, this improves the speed of the algorithm.threshold T is defined as formula (6).

$$T = \begin{cases} a, & t \le t_{ave} \\ a\sin\left(\frac{\pi}{2} \times \frac{t_{\max} - t}{t_{\max} - t_{ave}}\right), & t > t_{ave} \end{cases} \tag{6}$$

In formula (6), α is the threshold factor, its value is 0.3, t is the current evolution generations, t_{ave} is the average value of generations, $t_{\max}$ the maximum of generations.

2.2.3 Guo elite mutation operator and adaptive mutation probability
Mutation operator is a monoclonal antibody of mutation, it is used to produce changing on affinity values, and it is used to implement local search. In order to increase the selection pressure, accelerate the convergence rate, Guo Tao proposed elite parent Guo mutation operator[11].

Guo elite mutation operator is defined as formula (7):

$$m = \sum_{i=1}^{W} A_i x_i, \tag{7}$$

Where $\sum_{i=1}^{W} A_i = 1,\ -0.5 \le A_i \le 1.5$, x_i is the i-th antibody, m is the new generation of individual variation, W is the space of elite parent operator(In general, W=10%m),this makes the subspace of search algorithm can cover more space cover Multi-parent convex combination of space, and it ensures that the random search of ergodicity. This allows the individual to have more good gene propagation and survival opportunities, it can accelerate the convergence rate significantly.

Adaptive mutation probability p_m is associated with affinity values and evolution generation, it decreases with affinity values and evolution generation. p_m is defined as formula (8-10)：

$$p_m = \varphi p_{f_i} + (1-\varphi) p_{t_i} \tag{8}$$

$$p_{f_i} = f_i / \sum_{i=1}^{M} f_i \tag{9}$$

$$p_{t_i} = \exp[-t/\eta] \tag{10}$$

where p_{f_i} is the affinity probability of the i-th antibody, f_i is the value of affinity, p_{t_i} is the probability of evolutionary generation, t is the current evolutionary generations, φ, η are coefficient of impact factor.

2.2.4 Local Convergence Chaos Optimization Operator

Chaos is characterized by ergodicity, sensitive dependence on initial conditions and random-like behaviors, which is an aperiodic dynamics process, seeming disorderly and unsystematic; actually it contains order[12,13], when the algorithm gets into a local optimum, We use chaos optimization operator, it makes antibodies change, and it increase the diversity of the population, after several iterations, the Improved Hybrid Immune Algorithmn(IHIA) reaches a global optimal state. In this paper, we used Logistic mapping to calculate，one-dimensional Logistic map is a simple chaotic map[2], its mathematical expression is:

$$x_{k+1} = \mu . x_k (1 - x_k) \tag{11}$$

In formula（11），$x_k \in (0,1), k = 0,1,2,...,n$。When $3.56 \le \mu \le 4.0$，the Logistic mapping gets into a chaotic state, When the chaotic sequence is used in Improved Hybrid Immune Algorithmn(IHIA), the algorithm has strong global search capability, the convergence speed has also improved.

2.3 Simulation of IAHA algorithm

Step 1: Initialize population P, P is composed by the memory unit M and retain population P_r，$P = M \cup P_r$,when the system is initialized, P_r is randomly generated, and M=0, the number of antibodies of M is 30% of the total number of population.

Step 2:Evaluate the individual populations solution, calculate antibody affinity and concentration, electe M highest affinity antibodies, store them in memory.

Step 3: Clonal expansion

To ensure that large affinity and low concentrations of antibody with large clonal selection probability, antibody clones follow the formula (4),(5) and formula (6), each individual clone size is proportional to the degree of affinity, this is $N_c = k \times f_i / \sum_{i=1}^{n} f_i$，where n is scale of cloning individual, f_i is the affinity of individual i, k is the control coefficient.

Step4: Mutation

Monoclonal antibodies elite Guo Tao variation, the mutation operator is carried out according to formula (7). In early evolutionary of the algorithm, we use large affinity antibodies, this is to maintain the diversity of population, in the later stage of evolution, we use small affinity antibodies, small variations can improve the local fine-tuning ability, The mutation probability p_m is carried out according to formula (8).

Step 5: Updating of memory
Reselect the M high affinity individuals to build memory from mutated antibodies, at the same time, the parent individuals of low affinity will be eliminated, it regenerate next generation population P.

Step 6: Local optimization
Judge whether the algorithm gets into local convergence, if the algorithm gets into local convergence, the antibodies of the memory(40% of the total) performs the operation of local convergence chaos optimization, it is carried out according to formula (11),or it turns to step 2.

Step 7:Put evolution generation $t_{\max}$ as the termination condition, if it mets the condition ,then we stop this program and export the solution of this equation,or it turns to step 2.

RESULTS AND DISCUSSION

3.Experiment and Analysis
3.1 Parameter Analysis
(1).Size of antibody population。
The size of antibody population is generally taken from 50 to 150, for multimodal function, in order to find out extreme points as much as possible, we should take a big value, in this paper, it is taken as 200.
(2). The greater affinity, the greater the probability of low concentrations of antibodies are. in this paper, the affinity heuristic factor α is taken as 3,and the factor of concentration β is taken as 4.
(3). Number of clones
The number of clones has a greater impact on the convergence rate of algorithm, the larger the number, the faster the evolution algorithm is,in this paper, the cloning control coefficient K is taken as 25.
(4). Affinity values
The decoded function values are the affinity values of antibodies.
(5). Condition of termination: The maximum number of cycles is $t_{\max}$ =800.

3.2 Simulation results
In order to test the algorithm, we used six representative function from 13 standard test questions[14,15],and compared the result with the algorithms of reference [14] and reference [15].In table 1,the optimal value(f_{best}),the worst value(f_{worst}),the average value (f_{mean}),The shortest time to get the optimal solution(T/s),the power of the optimal solution(fn^{best}),the average of percentage error(σ) of 20 independent experiments on 4 algorithms are shown, where S_{Ti} is the i-th optimal path, S_0 is the known best value,

$$\sigma = \frac{\sum_{i=1}^{10} S_{Ti} - S_0}{10S_0} \times 100\%$$

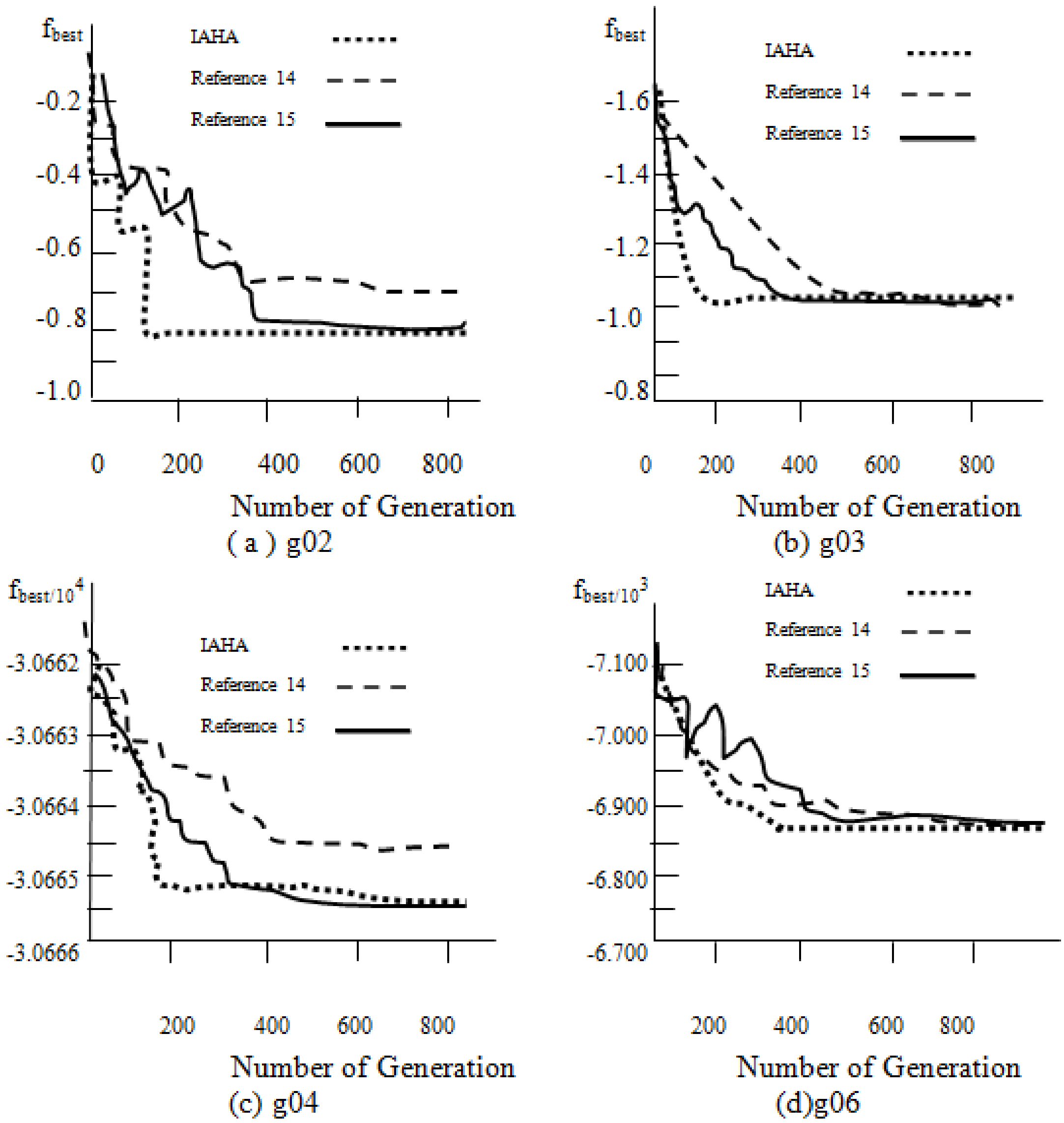
.

It can be seen from table 1 that, except for the detection function of g10, although we did not get the optimal value from, but the error between results and optimal value is small, for the σ, the results of this algorithm IAHA are less than the results of other two algorithms, for detection function g03 and detection function g06, the average of percentage error is 0,this means that the accuracy IAHA algorithm is high. Seen from the shortest time to get the optimal solution T(s), except for detection function g07, the shortest time to get the optimal solution, T(s) of IHIA is the shortest, this means that global convergence speed of the algorithm IHIA is fast. This shows that the proposed IAHA is an efficient algorithm for solving constrained optimization problems.

Table 1. The results of the six standard test function

Function S_0	Algorithm	f_{best} / f_{mean} / f_{best}	σ /%	T/s	f_n^{best}
g02 -0.803619	Reference 14	-0.271311/-0.371708/-0.754913	15.897	189.6	5
	Reference 15	-0.783613/-0.790628/-0.803619	2.897	85.9	11
	IAHA	-0.804091/-0.799097/-0.803619	0.911	56.9	16
g03 -1.00	Reference 14	-0.999/-1.000/-0.992	8.898	56.90	7
	Reference 15	-1.000/-1.000/-1.000	1.012	43.81	9
	IAHA	-1.000/-1.000/-1.000	0	30.65	20
g04 -30665.54	Reference 14	-30664.668/-30665.467/-30665.538	8.989	145.12	7
	Reference 15	-30663.190/-30664.730/-30665.460	3.231	90.71	9
	IAHA	-30665.938/-30665.541/-30665.54	1.781	65.78	13
g06 -6961.814	Reference 14	-6961.814/-6961.814/-6961.814	2.101	67.76	9
	Reference 15	-6961.814/-6961.814/-6961.814	0.21	51.45	7
	IAHA	-6961.814/-6961.814/-6961.814	0	45.67	14
g07 24.306	Reference 14	24.644/24.380/24.311	11.211	77.98	6
	Reference 15	24.709/24.463/24.306	1.675	71.89	12
	IAHA	24.321/24.311/24.306	0.767	72.56	9
g10 7049.25	Reference 14	9398.649/7509.321/7059.864	8.986	88.71	5
	Reference 15	7376.204/7293.503/7059.621	4.891	64.78	12
	IAHA	7147.785/7066.867/7049.202	1.832	56.8	13

Figure 1 and figure 2 show the evolution curves and processed curves of optimal solution from IAHA，reference 14 and reference 15.



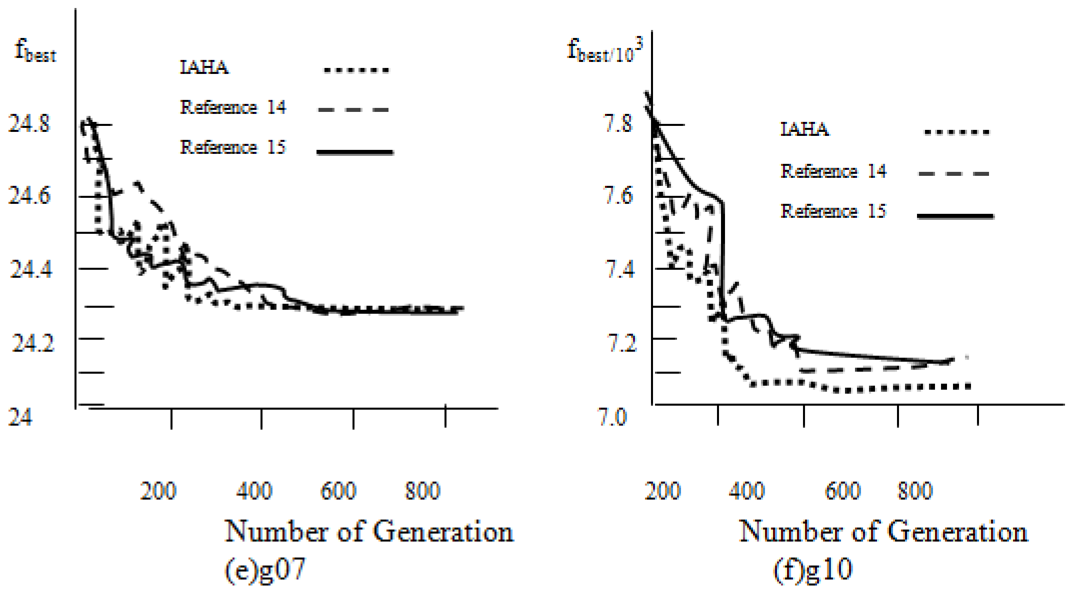


Figure 1. Evolution curve

It can be seen from figure 1 that, in the three algorithms, except the function g07, the stability of IAHA algorithm is stronger than the other two algorithms of reference [14] and reference [15] ; In all of the test functions, it has shown the characteristics of fast search.

For functions g02, g03 and g04, IAHA has reached the global optimum before iterative algebraic it iterative reaches 200 generations, on the other hand, the global optimal solution of IAHA better than the other two algorithms of reference [14] and reference [15].

For functions g02, the algorithm of reference [14] got the optimal solution after 370 generations (-0.7012.87), the algorithm of reference [15] got the optimal solution after 390 generations (-0.8028.45), the algorithm of IHIA got the optimal solution after 180 generations (-0.803612), this solution is close to the optimal solution (-0.803619).

For functions g04, the algorithm of reference [14] got the optimal solution after 410 generations (-30664.990), the algorithm of reference [15] got the optimal solution after 400 generations (-30665.238), the algorithm of IHIA got the optimal solution after 189 generations (-30665.530), this solution is close to the optimal solution (-30665.54).

For functions g06, the algorithm of reference [14] got the optimal solution after 408 generations (-6905.675), the algorithm of reference [15] got the optimal solution after 458 generations (-6961.814), the algorithm of IHIA got the optimal solution after 359 generations (-6961.814), this solution is close to the optimal solution (-6961.814).

For functions g07, the algorithm is not very stable in the early stages of the search, there is fluctuation in the optimal solution, but it is better than the other two algorithms of reference [14] and reference [15] in the late stages. the algorithm of reference [14] got the optimal solution after 40 generations (24.350), the algorithm of reference [15] got the optimal solution after 458 generations (24.310), the algorithm of IHIA got the optimal solution after 280 generations (24.308), this solution is close to the optimal solution (24.306).

For functions g10, the algorithm of reference [14] got the optimal solution after 400 generations (7156.897), the algorithm of reference [15] got the optimal solution after 410 generations (7180.765), the algorithm of IHIA got the optimal solution after 280 generations (7059.650), this solution is close to the optimal solution (7049.25).

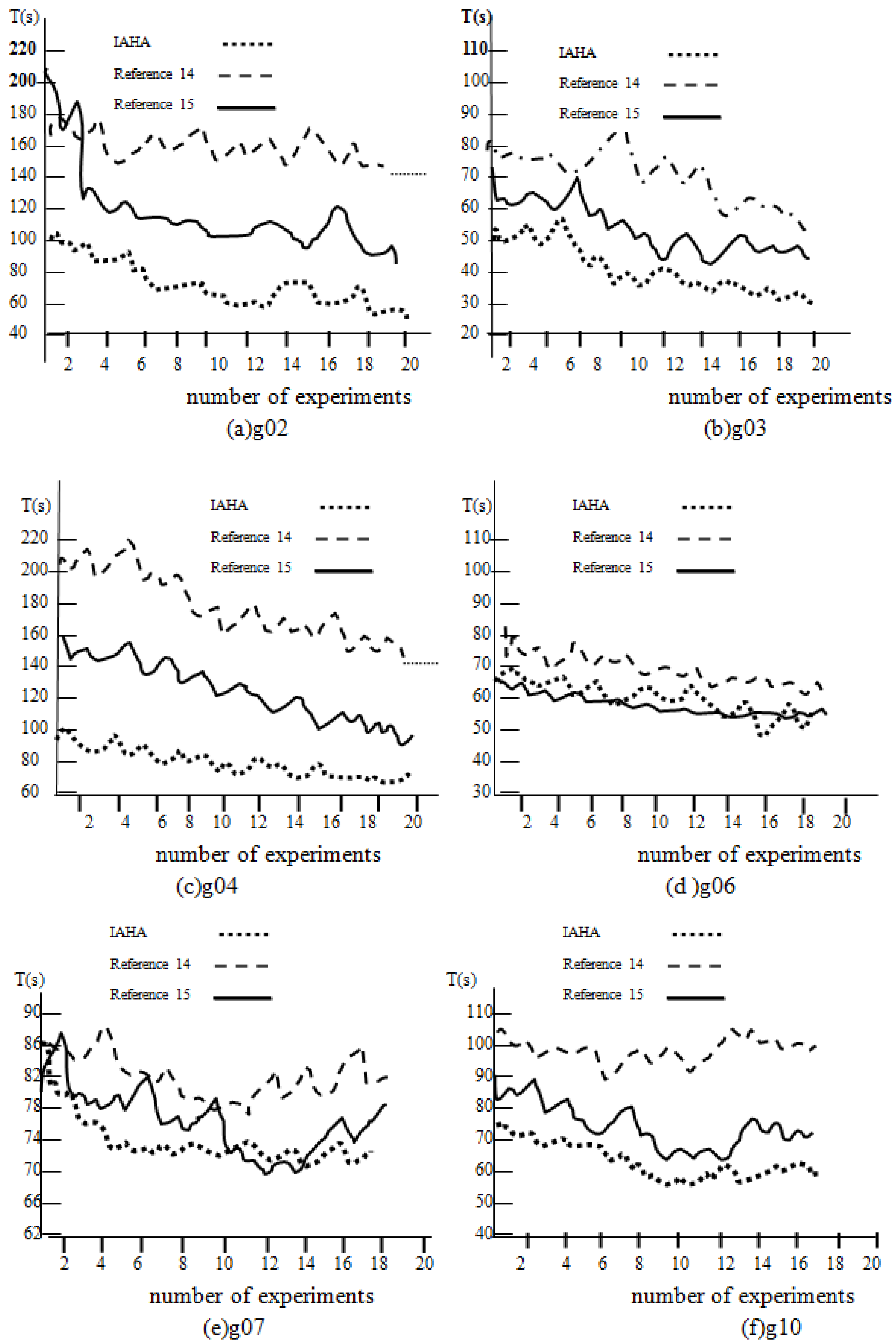


Figure 2. processed curves of optimal solution

It can be seen from figure 1 that, for the 20 experiments of function g02,g03,g04 and g10, the time to find optimal solution of IAHA is less than that of reference 14 and reference 15.

For functions g06, the time to find optimal solution of IAHA is less than that of reference 14,but the average time to find optimal solution of IAHA is more than that of reference 15,even so, the shortest time to find optimal solution of IAHA(45.67) is less than that of reference 14(67.76s) and reference 15(51.45s).

For functions g06, except in the 13th and 15th experiment, the time to find optimal solution of IAHA is more than that of reference 15,the time to find optimal solution and the average time to find optimal solution of IAHA in other experiment is less than that of reference 14 and reference 15. The shortest time to find optimal solution of IAHA T(s) is 72.56s, The shortest time to find optimal solution of reference 14 and reference 15 T(s) are 77.98s and 71.89s.

It can be seen from table 1,figure 1 and figure 2 that ,the IAHA algorithm is efficient.

CONCLUSION

(1).In this paper, we proposed an improved Hybrid Immune Algorithm(IHIA) about optimization problems of constrained function.
The characteristics of IHIA are : clonal selection operator is proposed, large affinity, low concentration of antibodies have large clonal selection probabilities, this improves the diversity of individual; Variable threshold selection operator is proposed, it Improves the speed of finding the optimal solution, and it avoids the algorithm to fall into local optimal solution.
(2). Guo elite mutation operator and adaptive mutation probability p_m are proposed in IHIA, they can improve local search speed;
(3). Local convergence chaos optimization operator is proposed, it makes algorithm get out from local optimization to global optimization.

Acknowledgements
This work was supported by Scientific Research Fund of Hunan Provincial Science Department under Grant (No:2014FJ4252), Educational Planning Research Fund of Hunan Provincial Education Department under Grant (No:XJK014CGD011， XJK014CGD013).

REFERENCES

[1] Zhang Zhuhong. *Applied Soft Computing Joural.***2007**,7(3):840-857.
[2] Zuo Xingquan, Mo Hongwei,Wu Jianping. *Informaition Sciences.***2009**,179(19):3559-3369.
[3] Guo Yinan,Wang Hui,Cheng Jian. *Acta Electronica Sinica.***2010**,38(4):966-970.
[4] Zhang Zhuhong,Qiang Shuqu. *PR&AI.***2007**,20(1):85-88.
[5] Wang Ling. Intelligent optimization algorithm with applications[M].*Beijing:Tsing University Press.***2001**:1-76.
[6] Zhou Ming,Sun Shudong.Genetic algorithm theory and applications[M].*Beijing: National Defense Industry Press.***1999**:5-134.
[7]Srinvas M,Patnaik L M. Adapvie. *IEEE Trans on Systems ,Man and Cybernetics.***1994**,24(4):656-667.
[8]Mori K,Tsukiyama M,Fukuda T. *Transaction-Institute of Electrical Engineers of Japan.***1993**,113C(10):872-878.
[9]Xiao Renbin,Wang Lei. *Chinese J .Computers.***2002**,25(12):1281-1290.
[10] Zhai Hongqun,Feng Maoyan. *Journal of Naning Normal University(Engineering and Technology Edition).***2011**,11(3):78-82.
[11]Guo Tao,Zhigniew Michalewicz. Inver-over operator for the TSP[C]. *Proceedings of the 5th International Conference on Parallel Problems Solving from Nature.***1998**:802-812.
[12]Li Bing,Jiang Wei-sun. *Control Theory &Application.***1997**,14(4):613-615.
[13] Mao Yongyi,Wang Yao. *Jounral of　Computer Applications.***2012**,32(10):2768-2770,2775.
[14]Shang Ronghua,Jiao Licheng,Ma Wengping. *Journal of Software* .**2008**,19(11):2943-2956.
[15]Yang Jian,Zhang Minhui. *Application Research of Computers.***2011**,28（11）:4029-4031.
[16]Yu Fan,Li Jixin. *Journal of Xi'an Technological University.***2014**,1(34):38-41.
[17]Zheng Tao,Pan Yumei. *Power System Protection and Control.* **2014**,42(1):77-83.