

Immune algorithm report

谌正/21721122

April 2018

1 Introduction

Artificial Immune Systems(AISs) represent a field of theories,principles,and concepts of modern immunology to design immune system-based applications in science and engineering [1].One role of the immune system is to protect the host organism against attacks from antigens and eliminate those cells that have been "infected".AISs are proving to be a very general and applicable form of bio-inspired computing.A great deal of work has gone into developing algorithms that extrapolate basic immune processes such as clonal selection,negative and positive selection,danger theory,and immune networks[4].To date,AIS have been applied to areas such as machine learning[8],optimization[6],bioinformatics[11],robotic systems[5],decision support systems[10],network intrusion detection[13],combinatorial optimization[2],scheduling[9],anomaly detection[14],fault diagnosis[7],computer security[[12],data analysis[3],and many other areas. This report is structured as follows:

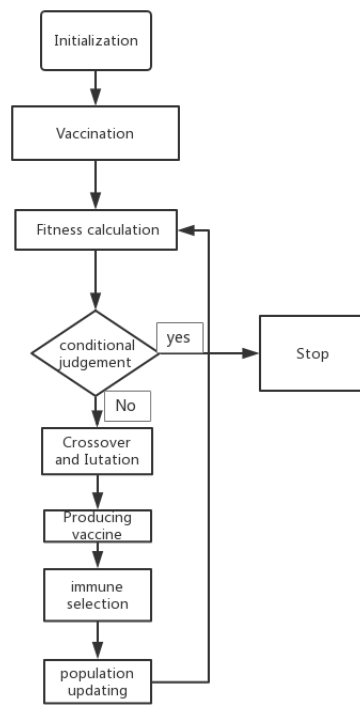
- 1.introduces a classic immune algorithm
2. for the problems existing in the classical immune algorithm, and improves it
3. for the combinatorial optimization problem: TSP problem, and proposes an immune model.

2 Immune Algorithm

2.1 Immune Operator

1.Full immunity refers to the type of immunity that each group of individuals in a group has an immune operation on every link after the action of genetic operators. It is mainly applied to the initial stage of individual evolution, but not in the process of evolution.

2.The target immunity refers to a type of immune response at the point of action only after a genetic operation.The target immunization is generally associated with the whole process of population evolution,and it is a basic operator of the immune operation.



2.2 The actual operation process of immune algorithm

- 1.The specific analysis of the solved problems(regarded as an antigen)
- 2.This feature information is processed to transform it into a solution to the problem(The set of solutions obtained according to the scheme is called the antibody based on the vaccine above)
- 3.Transform this scheme into an immune operator in an appropriate form

2.3 Vaccination

According to the prior knowledge, some genes in X are modified to make the individual have higher fitness for greater probability.

2.4 Immune selection

(1)Immunoassay:The detection of individuals vaccinated is not as good as the parent, indicating that the phenomenon of severe degeneration has occurred during the process of crossover and mutation, and this individual will be replaced by the corresponding individuals in the parent.

(2)Annealing selection: In the current progeny group $E_k = (x_1, \dots, x_{n_0})$, the individual x_i is selected by the probability $P(x_i)$ to enter the new parent group:

$$P(x_i) = e^{f(x_i)/T_k} / \sum_{i=1}^{n_0} e^{f(x_i)/T_k}$$

$f(x_i)$ is the adaptive degree of individual,
 T_k is a temperature control sequence approaching 0.

3 Immune Operator

3.1 The execution algorithm of immune operator

$a_{H,k}^i$ is the antibody obtained after vaccinating the k generation the i individual a_k^i . P_I is the probability of vaccinating individuals, and P_V is the probability of updating the vaccine. $V(a_k^i, h_j)$ is an inoculation operation modifying the gene on an individual a_k^i according to the pattern h_j , and N and m are the size of the population and the vaccine, respectively. $f(x)$ represent the fitness of individual x .

```
Begin:
Extraction of vaccine:
    Analyze the problem to be asked and collect the characteristic
information
    Estimation of patterns on specific gene sites based on char-
acteristic information:  $H = h_j | j = 1, 2, \dots, m$ 
     $k = 0$  and  $j = 0$ ;
    while(Conditions=true)
        if  $P_v = \text{true}$ , then  $j = j + 1$ ;
         $i = 0$ ;
```

```

for( $i \leq n$ )
  Vaccination: $a_{H,k}^j = V_{PI}(a_k^i, h_j)$ ;
  Immunoassay:if  $f(a_{H,k}^i) < f(a_{k-1}^i)$ , then  $a_k^j = a_{k-1}^i$ ; else  $a_k^i =$ 
 $a_{H,k}^i$ ;
   $i = i + 1$ ;
Annealing selection: $A_{k+1} = S(A_k)$ ;
 $k = k + 1$ ;
End

```

4 Improved Immune Algorithm

In the traditional artificial immune optimization algorithm, for cloning operator, the number of antibodies is preset, not determined by affinity. The mutation operator in the traditional artificial immune optimization algorithm is to conducting Gauss mutation at random. The improved immune optimization algorithm proposed in this paper improves the cloning and mutation steps of the traditional artificial immune algorithm, in which the clone number is proportional to the affinity of the parent antibody, and the mutation level of each antibody is inversely proportional to the affinity of the parent antibody in the mutation. At the same time, in the improved immune optimization algorithm, the excellent learning mechanism of the antibody is added, that is, the mutation process of each antibody will not only inherit the advantages of the parent antibody, but also learn from the antibody with the most affinity in the current population antibody. Therefore, the complete antibody clone equation can be described as follows:

$$h_i(t) = norm(Aff(Ab_i(t)))$$

$$C_i(t+1) = r_1 C_{max} H^n(t) + C_0$$

Among them, t represents the number of iterations. $norm()$ is a normalized function. $h_i(t)$ is the value of the normalized function, $Aff()$ is an affinity function of antibodies, C_{max} is the largest clone of antibodies, r_1 is a random number in $[0, 1]$, n is the exponent of power function, C_0 is the clone base of antibodies, $C_i(t+1)$ is the clone number of antibody $Ab_i(t)$, and $H^n(t)$ is antibody clone factor. when $n = 1$, the number of antibody clones is proportional to affinity. When $n > 1$, the greater the affinity, the faster the number of clones grows. When $n < 1$ the number of antibody clones with small affinity was larger. However, in the improved immune algorithm, choosing the appropriate power exponent n is very important for the size of antibody population and complexity of the algorithm. The relationship between the number of antibody clones and affinity of antibody is influenced by antibody clone factor.

In Gauss mutation, we use affinity based Gauss function to adjust the level of antibody variation. The antibody variation equation of the algorithm is as follows:

$$\Delta Ab_i(t+1) = \beta \gamma \exp(-h_i(t)/\eta)$$

Among them, the η is the control factor, the γ is the multiple of variation, the β is the random number in $[0, 1]$, and the $\Delta Ab_i(t+1)$ is the variation value between the parent antibody and the sub antibody, and the rest of the variables are the same as the equation (1). If η is greater, variation is higher; and vice versa. The control factor η has a big effect on efficiency of the evolution of the immune algorithm.

After learning, an evolved sub antibody not only carries the variation information from the parent antibody, but also focuses on learning the best antibodies in the group:

$$\Delta Ab_i^*(t+1) = \omega^*(Ab_g(t) - Ab(t))$$

ω is the random number in $[0, 1]$. $Ab_g(t)$ is the global best antibody. $\Delta Ab_i^*(t+1)$ is the learning information of $t+1$ generation. Weight coefficients k_1 and k_2 are used to balance the contribution of $\Delta Ab_i(t+1)$ and $\Delta Ab_i^*(t+1)$ to antibody evolution, respectively. In mutation operation, we need to skillfully treat excellent antibodies and non excellent antibodies. For excellent antibodies, excellent retention and affinity learning strategies are carried out according to affinity; and for non excellent antibodies, excellent learning strategies are also carried out in addition to excellent retention and affinity learning strategies. Therefore, the evolution of antibodies can be carried out according to the following formula.

When $Ab_g(t) = Ab_i(t)$

$$Ab_i(t+1) = Ab_i(t) + \Delta Ab_i(t+1)$$

When $Ab_g(t) \neq Ab_i(t)$

$$Ab_i(t+1) = Ab_i(t) + k_1 \Delta Ab_i(t+1) + k_2 \Delta Ab_i^*(t+1)$$

From the formula above, we know that the new candidate antibodies are composed of three items: the first is the parent antibody information, the second is the affinity learning information, and the third is the excellent learning information. In order to speed up the search speed of the algorithm, it is necessary to make a balance between increasing k_1 and reduce k_2 .

5 Immune algorithm model for TSP problem

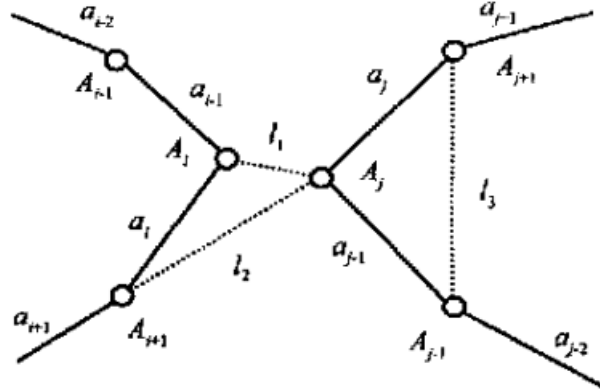
5.1 Analyze the problem to be solved and collect the characteristic information

Suppose a person starts from a city at a certain time and wants to go to the next target city. Generally, the first choice is the nearest city to the local route. If the target city is just a city ahead of the city, then, it will be replaced by the city with the smallest distance other than this city. This method is not a solution to the global problem, but in a very small range, such as the only three or four cities (relative to the global problem, which belongs to a local problem), this consideration is often a better strategy. Of course, whether it can be the final solution still need further judgment.

5.2 Making immune vaccines based on characteristic information

Based on the above understanding, in terms of the characteristics of the TSP problem, in the final solution, that is the selection of the best path, including and in a large degree, includes the shortest distance between adjacent cities. This characteristic of the TSP problem can be used as a characteristic information or knowledge for the solution of the problem, so that it can be viewed as a reference. In order to extract the vaccine from the problem, in the specific implementation process, it is only necessary to use a general cyclic iterative method to find out the neighboring cities of all cities (of course, a city may be a neighboring city of two or more cities, and probably not. " The vaccine is not an individual, so it can not be used as a solution to the problem. It only has the characteristics of individuals in some gene locations.

5.3 Vaccination



Generally, suppose that, in the TSP problem, all cities nearest to A_i are A_j , and the two part is not directly connected, but is in the two segment of a certain path

$$A_{i-1} - A_i - A_{i+1} \text{ and } A_{j-1} - A_j - A_{j+1}.$$

As shown in the figure above, The current traversal path is

$$\pi = A_0, \dots, A_{i-1}, A_i, A_{i+1}, \dots, A_{j-1}, A_j, A_{j+1}, \dots, A_N$$

its corresponding path length is:

$$D\pi = \sum_{k=1}^{i-1} a_k + a_i + \sum_{k=i+1}^{j-2} a_k + a_{j-1} + a_j + \sum_{k=j+1}^N a_k.$$

Under the condition of the immune probability P_i , for the city A_i , the immune operator will arrange its adjacent city A_j into his next target city, then the original traversal path is adjusted as follow:

$$\pi = A_0, \dots, A_{i-1}, A_i, A_j, A_{i+1}, \dots, A_{j-1}, A_{j+1}, \dots, A_N$$

then the length of the path is adjusted as follows

$$D_{\pi_c} = \sum_{k=1}^{i-1} a_k + l_1 + l_2 + \sum_{k=i+1}^{i-2} a_k + l_3 + \sum_{k=j+1}^N a_k$$

Because A_j is the nearest point from A_i in all cities, l_1 must be the shortest or the sub short side in the triangle made up of $A_i - A_j - A_{i+1}$ (at this time l_2 must be the shortest side. for the reason that if $a_i < l_1$, then the most nearest city from A_i will be A_{i+1} rather than A_j); However, between A_{j-1}, A_j, A_{j+1} , this property does not necessarily exists, so in most cases, the reduction of l_3 to $a_{j-1} + a_j$ is greater than the increasement of $l_1 + l_2$ to a_i . In this local environment, the operator makes a best adjustment to the path. Of course, whether this adjustment can contribute to the whole path still needs further study. However, it is not difficult to make the following relationship from the analysis process:

$$P(D_{\pi_c} < D_{\pi}) \gg P(D_{\pi_c} > D_{\pi})$$

The so-called "adjustment" process is immune injection process based on a specific vaccine when solving the TSP problem.

References

- [1] Y. Cao D. Dasgupta and C. Yang. An immunogenetic approach to spectra recognition London, U.K. in *Proc. Int. Conf. Genetic and Evol. Comput. Conf., Orlando, FL*, 1999.
- [2] D. Dasgupta. An artificial immune system as a multiagent decision support system. *IEEE Int. Conf. Syst., Man, Cybern., San Diego, CA, Oct*, 32(3816–3820), 1998.
- [3] D. Dasgupta. Artificial Immune Systems and Their Applications Berlin, German. *Springer-Verlag*, 1999.
- [4] L. N. de Castro and J. Artificial Immune Systems: A New Computational Intelligence Paradigm London, U.K. *Springer-Verlag*, 2002.
- [5] L. N. de Castro and F. J. V. Learning and optimization using the clonal selection principle London, U.K. *IEEE Trans. Evol. Comput*, 6(239–251), 2002.
- [6] T. Fukuda and M. T. K. Mori. Parallel search for multimodal function optimization with diversity and learning of immune algorithm Germany: Springer-Verlag. *Springer-Verlag*, 1999.
- [7] E. Hart and P. Ross. The evolution and analysis of a potential antibody library for use in job-shop scheduling London, U.K. in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds, 1999.
- [8] J. E. Hunt and D. E. Cooke. Learning using an artificial immune system- London, U.K. *Springer-Verlag*, 19(189–212), 1996.
- [9] J. Kim and P. Bentley. Towards an artificial immune system for network intrusion detection: An investigation of clonal selection with negative selection operator. in *Proc. IEEE Int. Congr. Evol. Comput., Seoul, Korea*, 2001.

- [10] G. Nicosia. Immune algorithms for optimization and protein structure prediction Catania, Italy. *Math. Comput*, 2004.
- [11] R.Konig and T.Dandekar. Improving genetic algorithms for protein folding simulations by systematic crossover. *Biosystems*, 50(17-25), 1999.
- [12] A. S. Perelson S. Forrest. Self-nonsel self discrimination in a computerLondon, U.K. *in Proc. IEEE Symp. Research in Security and Privacy, Oakland, CA*, 1994.
- [13] G. Narzisi V. Cutello and G. Nicosia. A multi-objective evolutionary approach to the protein structure prediction problem. *J. Royal So. Interface*, 3(139–151), 2006.
- [14] G. Nicosia V. Cutello and M. Pavone. A hybrid immune algorithm with information gain for the graph coloring problemChicago. *in Proc. LNCS on Genetic and Evol. Comput. Conf*, 2723(171–182), 2003.