Scientific Research Publishing

Journal of Computer and Communications, 2019, 7, 166-174
http://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

An Improved Immune Algorithm for Solving Path Optimization Problem in Deep Immune Learning of Gene Network

Tao Gong, Mengyuan Wang

College of Information Science & Technology, Donghua University, Shanghai, China
Email: taogong@dhu.edu.cn

How to cite this paper: Gong, T. and Wang, M.Y. (2019) An Improved Immune Algorithm for Solving Path Optimization Problem in Deep Immune Learning of Gene Network. *Journal of Computer and Communications*, **7**, 166-174.
https://doi.org/10.4236/jcc.2019.712016

Received: December 14, 2019
Accepted: December 27, 2019
Published: December 30, 2019

Abstract

In order to overcome some defects of the traditional immune algorithm, the immune algorithm was improved for solving a path optimization problem in deep immune learning of a gene network. Firstly, the diversity of the solution population was enhanced in the evolution process by improving the memory cell processing method. Moreover, effective gene information was dynamically extracted from the genes of the excellent antibodies to make good vaccines in the process of immune evolution. Worse antibodies were optimized by vaccinating these antibodies, and the convergence of the immune algorithm to the optimal solution was improved. Finally, the feasibility of the improved immune algorithm was verified in the experimental simulation for solving the classic NP problem in deep immune learning of the gene network.

Keywords

Improved Immune Algorithm, Path Optimization, Memory Cell Processing, Vaccine

1. Introduction

The biological immune system is a complex distributed information-processing/learning system with good inherent advantages such as diversity, adaptability, immune tolerance, and immune memory. Inspired by the biological immune system, intelligent optimization algorithms simulate the complex information processing mechanisms of immune system, such as identification, learning, memory and etc. The robustness of the immune algorithm is better in solving the distributed complex problems, with better intelligence and better convergence to the global optimal solution in the solution-searching procedure [1]. Immune algorithms have been used in many fields such as machine learning, intelligent control, intrusion detection, and function optimization.

In the deep immune learning of gene network, there is a similar NP problem like the traveling salesman problem (TSP). Many intelligent algorithms such as genetic algorithm [2], particle swarm algorithm [2], neural network, immune algorithm, ant colony algorithm were used to solve the traveling salesman problem [3]. Wang Lei *et al.* designed an immune genetic algorithm for solving TSP problems by adding immune operators such as immune vaccine and immune selection into the genetic algorithm [4]. A method for extracting vaccines from best antibodies among the father antibodies was studied [4] [5]. The shortest path between the city and its nearest neighbor was utilized as a vaccine [6] [7]. In order to improve the immature convergence of the genetic algorithm, the improved vaccination strategy was designed to improve the global search efficiency [7]. The dynamic vaccine extraction strategy was adopted to improve the immune algorithm for solving the TSP problem [8].

In the traditional immune algorithm, the objective function for solving the problem was regarded as the antigen of foreign invasion, and the antibody generated in the immune response was regarded as the solution of the problem [9] [10] [11] [12]. Therefore, the evolution & maturation mechanism of different affinity antibodies is the process of searching the optimal solution of the objective function with the following steps, as shown in **Figure 1**.

Step 1: Antigen production: A description of the objective function and constraints of the optimization problem.

Step 2: Initial antibody production: The initial antibody population was generated randomly.

Step 3: Fitness calculation of the antibodies: calculate the affinity or fitness of each solution in the solution group, and provide a basis for the generation of memory cells in the future.

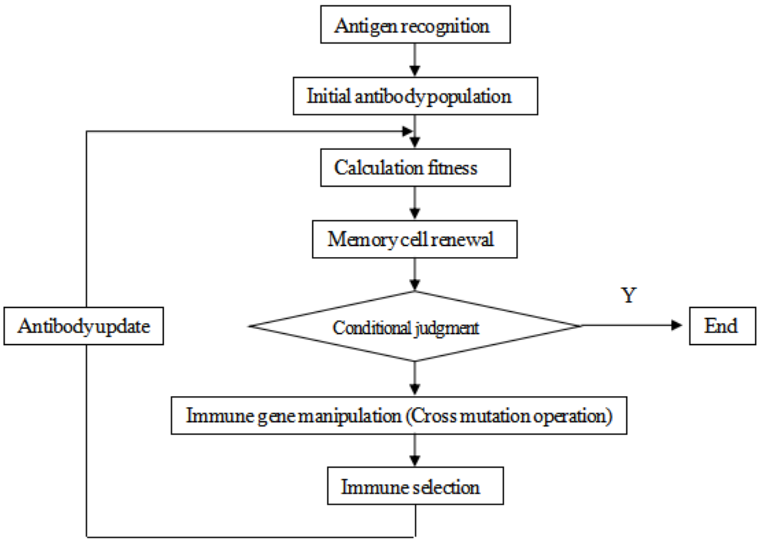

Figure 1. Workflow of a traditional immune algorithm.

Step 4: Memory cell production: sort according to the degree of affinity, select individuals with higher affinity to form a new group, and take the first few antibodies to form a memory.

Step 5: Judgment of termination conditions: Determine whether the termination condition is met (find the optimal solution or the maximum number of iterations is reached), then the algorithm ends; otherwise, continue.

Step 6: Immune Gene Manipulation: Evolutionary updates of antibodies using crossover and mutation operators.

Step 7: Immune selection operation: The antibody after the presence of the gene is preferentially selected, and the antibody with high affinity is retained.

Step 8: Antibody group updating: A new antibody is randomly generated to form a new antibody group with a high affinity antibody in the original antibody group, and the process returns to step 3.

2. Improved Immune Algorithm

2.1. Improvement of Memory Cells

In the past, one or several antibodies with the highest affinities were used as memory cells by a traditional immune algorithm. This method often retained a relatively similar antibody population, which made many important immune information lost and might decrease the diversity of memory cells. However, the diversity might determine the future evolutionary direction of the antibody population and was used to avoid the precocity of the local optimum. Therefore, it is necessary to improve the immune algorithm to save enough immune information and ensure the diversity of memory cell information. Thus, the traditional immune algorithm was improved to keep certain distances among memorized antibodies and select the antibodies with high affinities (not only the highest ones) as the memory cells.

The theory of information entropy or the Euclid distance between two antibody coding strings were commonly used as the antibody similarity judgment methods. However, these methods only judged the genes with the same gene positions in an antibody, and couldn't represent the relative positional relationship of the gene arrangement. Therefore, the traditional methods were not good at judging the similarity among the antibody-solutions of the path optimization problem. In the improved immune algorithm, in consideration of the relative position, it was judged whether or not the antibodies were similar according to whether or not the length of the same gene fragment between the antibodies exceeded the threshold.

2.2. Vaccine Improvement with Vaccine Dynamic Extraction and Vaccination Strategy

With the real number coding, the dynamic vaccine extraction and vaccination strategy were proposed, in order to form a vaccine library by dynamically extracting excellent gene fragments as vaccines in a memory cell bank of excellent antibodies. Then, in the process of immune gene manipulation, the vaccine was vaccinated with a certain probability and combined with the cross mutation procedure to update the antibodies with poor affinities. In a better way, this improved strategy retained the excellent genes and expanded the search space of the algorithm to increase the search ability of the algorithm, while maintaining the diversity of the algorithm and avoiding the possibility that the algorithm falls into any local optimum.

Vaccines were utilized to activate the production of useful antibodies, which were useful in eliminating viruses and defending against invasions into the body in time. The extraction of the vaccine in the traditional immune algorithm was based on the basic feature extraction from the prior knowledge of the target problem and the valid information extraction from the genes of the best individuals in the process of population evolution. The acquisition of the vaccine ensured that the antibody retained excellent characteristics during the mutation process, so the proportion of the antibody genes were effectively optimized and the antibodies evolved rapidly.

In the improved immune algorithm, m antibodies ($m < N$, N represents the number of memory antibodies in the memory antibody group) were randomly selected from the memory antibody group $M(t)$, and the common gene fragment of m antibodies was recorded as one vaccine in the vaccine library $V(t)$, which retained the excellent gene fragment of the memory antibody. With the evolution of the population, dynamic updating of the vaccine library for vaccine extraction better preserved the excellent antibody gene and accelerated the convergence speed of the immune algorithm.

In the traditional immune algorithm, the vaccination operation was before the cross-variation operation, and the partial antibodies were randomly vaccinated. However, when the vaccine immunization operation was insufficient, the loss of diversity was accelerated and the premature convergence of the traditional immune algorithm occurred. Moreover, the traditional crossover operator randomly selected the intersection position, and this operation might destroy the structures of other excellent genes, after a large number of crossover operations. This operation might generate a large number of infeasible offspring individuals, which resulted in the premature convergence or redundant iteration of the traditional immune algorithm. In the improved immune algorithm, unsuitable antibodies were vaccinated by the probability-based crossover operation to optimize the antibody population. This improved crossover operation better retained the excellent genes and accelerated the optimal-solution searching of the immune algorithm. By combining the mutation operation, the diversity of the improved immune algorithm was guaranteed, and the prematurity of the local optimum was prevented in the searching procedure of the improved immune algorithm.

In the traditional immune algorithm, the vaccination operation was performed by random injection, which was not conducive to the optimization of antibodies, because the location of random injections might easily destroy the sequences of the original antibodies. Since each vaccine itself already contained the location information, the location information was injected by the improved immune algorithm at the time of vaccination, so that the positions of the injected vaccines were more reasonable and useful.

3. Improved Immune Algorithm for Solving Path Optimization Problem in Deep Immune Learning of Gene Network

3.1. Problem Representation

Like the traveling salesman problem, the shortest path optimization problem in the deep immune learning of gene network was represented in **Figure 2**. Given n relative nodes in the gene network graph, the shortest path was found to generate new pathway for reactivating the immune response against such non-self as tumors, as described mathematically below.

$$\min \quad f = \sum_{i=1}^{n-1} d(c_i, c_{i+1}) + d(c_n, c_1) \tag{1}$$

Here, $st.\ c_i \in C, C = \{c_1, c_2, c_3, \ldots, c_n\}, i = 1, 2, 3, \ldots, n$, C denoted the node collection, c_i represented the number of the node.

3.2. Algorithm Design and Implementation

In order to evaluate the merits of individuals, the affinity was calculated below.

$$aff(b) = 75.6 \times L \times \sqrt{n} / length(b) \tag{2}$$

Here, L denoted the side length of the smallest square containing all nodes, n denoted the number of nodes, and $length(b)$ denoted the path length corresponding to the antibody.

In the improved immune algorithm, suitable distance between one memory antibody and another was kept, and the suitable antibody with the highest fitness was saved as the memory cell. For example, among the antibody populations, the antibody A with the highest retention affinity was a memory cell. Afterwards, the antibody B with the second highest affinity was compared with the comparative antibody A. When the same gene fragment of A and B exceeded the certain

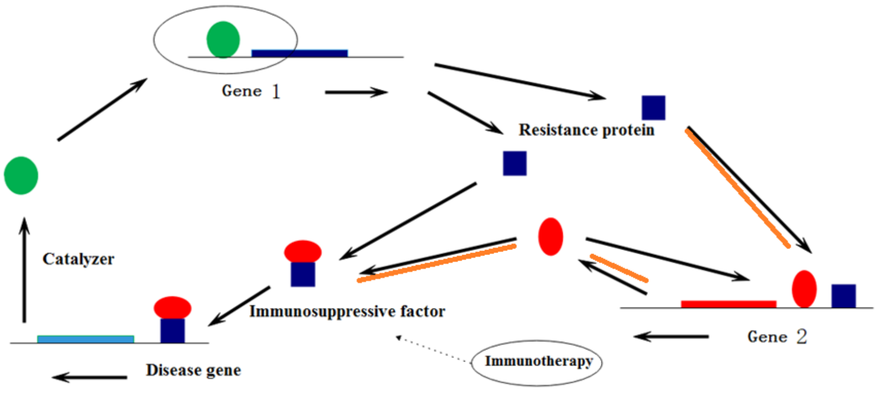

Figure 2. Path planning problem of the gene regulatory network.

threshold, which was defined as 80% of the length of the antibody, the antibodies *A* and *B* were regarded as similar, and the antibody *B* wasn't recorded as a memory cell. Otherwise, the antibody *B* was saved as a memory cell. Furthermore, the antibody *C* with the third highest affinity in the antibody group was compared with the remaining memory antibodies to determine whether there was a same gene fragment exceeding the threshold. If there was not any same gene fragment exceeding the threshold, the memory antibody group was added. By analogy, a memory antibody population $M(t)$ of N memory antibodies was generated.

In the improved immune algorithm, the two parts of the antibody population were processed by two steps of immunological manipulation.

1) The partial neighborhood search strategy was used for the high affinity anterior NP/4 antibodies in the optimization.

If a large number of random cross-mutation operations were performed on antibodies with high affinity, the orders of the excellent genes in these antibodies would be destroyed, which lead to the premature convergence or redundant iteration of the algorithm and then failed in finding the optimal solution. The local neighborhood search strategy was useful for better preserving the good genes, which was helpful to further accelerate the convergence of the improved immune algorithm and shorten the running time of searching the optimal solution. This operation was designed to generate a new local domain progeny individual by adopting a right neighbor gene exchange method for each gene in the antibody. For example, the antibody $S = (1, 6, 5, 4, 2, 3)$ exchanged gene 1 with its right domain gene to generate a new individual $S' = (6, 1, 5, 4, 2, 3)$.Therefore, the local neighborhood searching operation generated some local neighborhood progeny individuals, and selected the neighbor individuals with the greatest fitness to compare with the matrix. Afterwards, the higher affinity was used to update the matrix.

2) The proposed dynamic vaccine strategy was employed for 3NP/4 antibodies with worse affinity in the antibody population.

In the cloned antibody, the vaccine was optimized by the probability of Pv to optimize the antibody. And the search space was expanded by a random multiple-point two-point crossover operation to ensure the diversity of the population and to find better antibodies. The vaccination was performed according to the position of the first gene of the vaccine in the original antibody, and the part of the original antibody that conflicted with the vaccine was deleted. The specific operation flow chart is shown in **Figure 3**.

The immune selection operation was designed according to the natural selection mechanism, by retaining excellent antibodies and optimizing the population. In the improved immune algorithm, two parts of the immune gene operations were compared with each other.

4. Experimental Results and Analysis

In order to test the effectiveness of the proposed algorithm, 31 nodes were used in the experiments of the algorithm verification. The improved immune algorithm (IIA) was compared with the general immune algorithm (GIA) and the shortest path vaccine based immune algorithm (SPV-IA). In the Matlab-based simulations, the population size was set with n = 200 in the three algorithms, and the number of iterations was set with gen = 1000. After each algorithm was run for 10 times, the results were compared in **Table 1**. The most satisfactory path diagrams of the three algorithms after running 10 times are shown in **Figures 4-6**.

In **Table 1**, the most satisfactory value of the improved immune algorithm was 15,398.3952, which was obviously superior to those of both the algorithms GIA and SPV-IA. The average value of the optimal solutions of the improved immune algorithm was the smallest, so the improved immune algorithm performed more smoothly than the others. Moreover, the improved immune algorithm had less population dispersion and better convergence to the optimal solution for solving the classic NP problem in deep immune learning of the gene network.

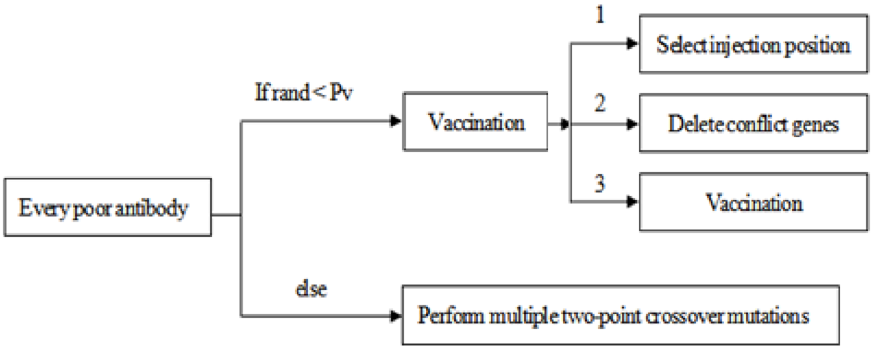

Figure 3. Flow chart of poor affinity antibody gene operation.

Table 1. Experimental comparison of the improved immune algorithm with others.

Generation maximum	GIA	*SPV-IA*	Improved immune algorithm
1	15,539.3995	16,072.8857	15,820.5409
2	16,554.6716	15,896.2177	15,799.7141
3	15,896.3375	16,314.9262	15,928.5002
4	16,035.7134	15,931.7508	15,422.2810
5	15,789.3462	15,623.2187	15,422.2810
6	15,796.7793	15,709.3367	15,398.3952
7	16,503.4526	15,836.0909	15,422.2810
8	15,605.8659	15,536.9961	15,771.6671
9	16,867.4302	16,390.5191	15,398.3952
10	16,445.5187	15,760.9406	15,475.2496
Mean	16,103.45149	15,907.28825	15,585.93053
Most satisfactory value	15,539.3995	15,536.9961	15,398.3952
Most dissatisfied value	16,867.4302	16,390.5191	15,928.5002
Worst value	1328.0307	853.523	530.105

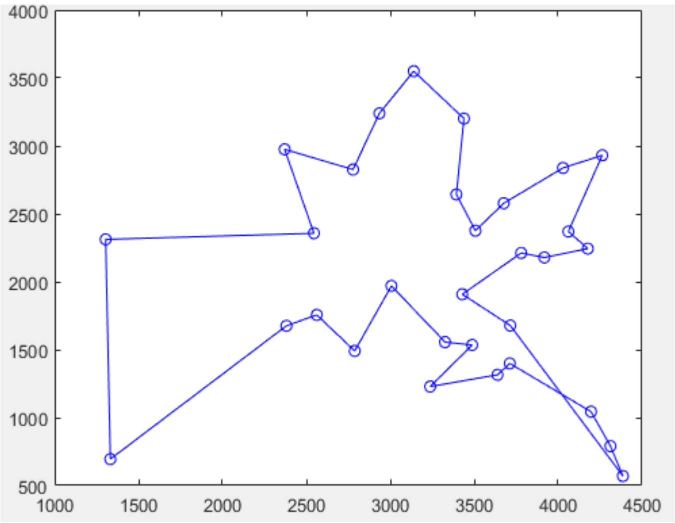

Figure 4. The most satisfactory value path for the GIA.

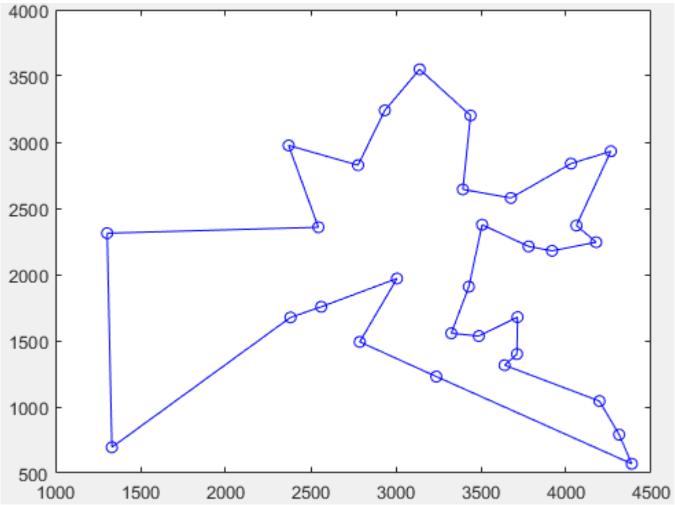

Figure 5. The most satisfactory value path for the SPV-IA.

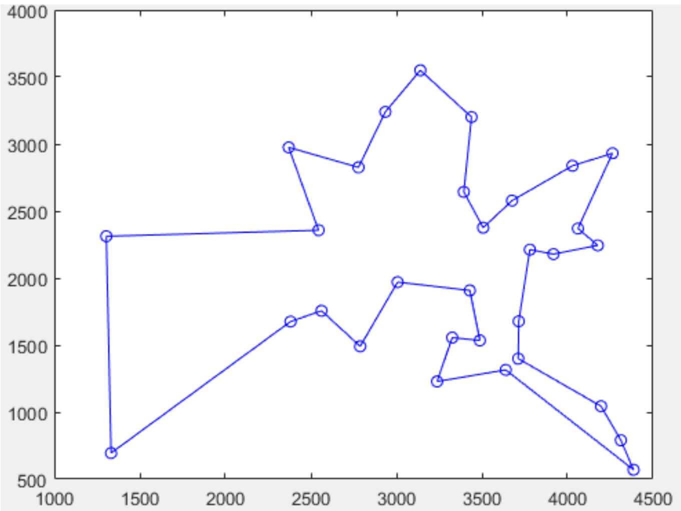

Figure 6. The most satisfactory value path for the IIA.

5. Conclusion

To overcome the shortcomings of traditional immune algorithms for solving the classic NP problem in deep immune learning of the gene network, the treatment of memory cells and the strategies for dynamic vaccine extraction & vaccination were improved in the new immune algorithm. The improved immune algorithm achieved good results in solving the classical NP problem in deep immune learning of gene networks, in comparison with the other algorithms GIA and SPV-IA.

Acknowledgements

This research was supported by the project grants from National Natural Science Foundation of China (Grand No. 61673007).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Cai, Z.X. and Gong, T. (2004) Advance in Research on Immune Algorithms. *Control and Decision*, **19**, 841-846.

[2] Zhou, H., Song, M. and Pedrycz, W. (2018) A Comparative Study of Improved GA and PSO in Solving Multiple Traveling Salesmen Problem. *Applied Soft Computing*, **64**, 564-580. https://doi.org/10.1016/j.asoc.2017.12.031

[3] Liu, Z.H., Zhang, Y.J., Zhang, J. and Wu, J.H. (2010) Using Combination of Ant Algorithm and Immune Algorithm to Solve TSP. *Control and Decision*, **25**, 695-700, 705.

[4] Wang, L., Pan, J. and Jiao, L.C. (2000) Evolutionary Algorithm Based on Immune Strategy. *Natural Science Progress*, **10**, 451-455.

[5] Wu, Y.T. (2011) Solution of TSP Problem Based on Immune Algorithm. *Science and Technology Information*, No. 36, 451-452.

[6] Wang, L., Pan, J. and Jiao, L.C. (2000) Immunization Planning. *Journal of Computers*, **23**, 806-812.

[7] Sun, X.L. and Shao, D.H. (2007) Immune Algorithm Based on Vaccination. *Computer Engineering and Design*, **28**, 3960-3962.

[8] Yan, Y.T., Liu, F. and Jiao, L.C. (2008) Dynamic Vaccine Strategy for Immune Algorithm for Solving TSP Problem. *Journal of Xidian University*, **35**, 37-42.

[9] Kong, L.Y. (2015) Improved Immune Genetic Algorithm for the Optimal Path of Traveling Salesman Problem. *Journal of Mathematics*, No. 2, 361-367.

[10] Pan, G., Li, K.L., Ouyang, A.J. and Li, K.Q. (2016) Hybrid Immune Algorithm Based on Greedy Algorithm and Delete-Cross Operator for Solving TSP. *Soft Computing*, **20**, 555-566. https://doi.org/10.1007/s00500-014-1522-3

[11] Zheng, J.Q., Chen, Y.F. and Zhang, W. (2010) A Survey of Artificial Immune Applications. *ArtifIntell Rev*, **34**, 19-34. https://doi.org/10.1007/s10462-010-9159-9

[12] Yang, J. and Zhang, M.H. (2011) Novel Immune Algorithm for Solving Constrained Optimization Problems. *Application Research of Computers*, **11**.