International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P), Volume 10, Issue 6, June 2020

An artificial immune algorithm for optimisation

Nguyen Nhu Trang, Doan Thi Minh Thai

***Abstract-* Various algorithms inspired by evolutionary and physical processes have been extensively applied in solving complex optimisation problems. Immune inspired algorithms are recently used for optimisation ones. In this paper, we impove an immune algorithm to solve travelling salesman problem (TSP). The effectiveness of the proposed algorithm is investigated through experiments on some datasets. Experimental results show that the studied algorithm outperforms Genetic Algorithm.**

***Index Terms*-Optimisation, genetic algorithms, immune algrithms, CLONALG.**

I. Introduction

There are a lot of models proposed for optimisation. Most of them are based on computational intelligence approaches like Genetic Algorithm (GA), Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO), etc. A subfield of computational intelligence, Artificial Immune System (AIS) is consider as one of the most promising approach for optimisation. AIS is computational systems inspired by theoretical immunology, observed immune functions, principles and mechanisms in order to solve many problems [7].

The Clonal Selection Algorithm (CLONALG) belongs to the field of AIS. It is related to other Clonal Selection Algorithms such as the Artificial Immune Recognition System [2]. We would like to improve CLONALG for TSP problem in this research.

The TSP can be declared informally: A tourist wants to visit n different cities. Starting from a certain city, tourists will go to the remaining cities, each city comes exactly once, then back to the first city. Find your journey in such a way that the total cost is minimal.

The rest of the paper is organized as follows. In the next section, we present the background of immune and improve CLONALG algorithms. Section 3 presents our experiments and comparision with GA. Section 4 concludes the paper and discusses some possible future works.

Nguyen Nhu Trang, Thai Nguyen University of Medicine and Pharmacy, Thai Nguyen City, Vietnam.

Doan Thi Minh Thai, Thai Nguyen University of Education, Thai Nguyen City, Vietnam.

II. CLONALG Algorithms

A. Immune-inspired idea for optimisation

The immune system is a biological system that protects the body against constant attacks from organisms from outside. More specifically, it is the process by which the body produces antibodies (Antibody - Ab) that fight off external antigens (Antigen - Ag). Thus, whenever the antigen enters the body, the immune system is tasked with finding and producing the appropriate antibodies to identify and eliminate the antigen. Based on this idea, we can build an optimal problem model based on the immune system as follows:

- Each alternative is an Ag (Antigen) antigen
- Space X is the space of the whole antibody (Antibody - Ab) of the immune system including antibodies with the combination of Ag antigen and also those that do not have the ability to identify Ag antigens.
- C is the set of Ab antibodies capable of identifying Ag antigens
- The overall optimal solution x* is that Ab* is most suitable for Ag on the set of determination C

B. Improved CLONALG

A modified version of the CLONALG in [2] is showed in the following algorithm.

Algorithm: Optimal CLONALG (CLONALG)

Step 1. Initialize a random population of size N0, initialize the M2 memory set to be ∅

Step 2. Calculate the values for the elements in the P population and arrange the elements in the ascending order of the function values, select the best n1 elements (the first n1 elements) in the P population P1 and scaling up by the value of each of these elements (smaller values are produced more) are populations C.

Step 3. Mutation of the elements in C population in proportion to the value of the element (the smaller the mutation the smaller rate is) is the population C1.

Step 4. Calculate the function value for the elements in the population C1 and arrange the elements in the ascending order of the function values obtained by population M1

Step 5. Select the best n2 elements (the first n2 elements) in M1 and add them to the M2 memorization set.

Step 6. Calculate the function value for the elements in M2 and sort the elements in ascending order of values.

Step 7. If the number of elements in M2 is greater than n3 then M2 ← select(M2, n3), choose to get the best n3 elements (first n3 elements) retained in M2.

Step 8. Creating populations M: M← P + M1 + M2.

Step 9. Calculate the path value for the elements in M and sort the elements in ascending order of the path values.

Step 10. If the size of M is larger than n4-n5 then retain the best n4-n5 elements (first n4-n5) in M2 (M2 ←select (M2, n4-n5))

Step 11. Initialize a random population of size R n5

Step 12. Redefine the population P: P ← M + R (P←insert(M, R)). Population size P: size (P) = size (M) + size (R) ≤ (n4-n5) + n5 = n4

Step 13. Repeat Step 2 until the end conditions are met: The termination condition may be after a certain number of iterations, or after a specified number of smallest values in constant M2 (the function value of the first element in constant M2)

Step 14. X ←select (M2,1) is .

III. EXPERIMENTS

A. Compare population control methods between AIS and GA

The method of selecting elements of AIS and GA is similar, the selection of elements is based on their suitability, whichever is more appropriate, the probability of selection will be higher.

Replication method: After selecting good elements, AIS performs replication but there is no replication in GA. AIS replication is based on the appropriateness of the elements, the higher the suitability of the element the greater its number of copies.

After GA has selected the elements, and AIS has selected the elements and replicated these elements, both GA and AIS perform the next process of transforming these elements. GA performs transformations with two operations: Crossover and Mutation, AIS performs transformation with the only operation is mutation.

Hybridization is only available in GA and this is a major contribution to the transformation of the average hybridization rate of 0.6

The mutation in GA is applied with a very low rate of about 0.001, and the mutation rate on every selected element is the same. This is different from AIS, in AIS only the transformation for the only individual is a 100% mutation, the AIS mutation for each element is inversely proportional to its suitability, the more appropriate The higher the value (the smaller the function value), the smaller the mutation rate.

The method of selecting the elements of the old generation to the new generation both AIS and GA are available.

The method of element elimination is available in both GA and AIS, but in GA, the element removal depends only on the interaction of the solution and its adaptive function without the interaction between the solutions. AIS's immune network mechanism.

The method of creating good memory set is only available at AIS, GA only remembers the best element.

The method of generating a new random population is unique to AIS.

The method of building the population for the next generation is also different. In GA, the population of the next generation consists of a number of elements of the old generation, elements created during selection and transformation. In AIS, the population of the next generation consists of a number of elements of the previous generation, elements created in the selection, replication and transformation process, elements in the memorization set and newly added elements randomly generated.

B. Assessment of convergence and population diversity of AIS and GA

Because of the different methods of population adjustment, the convergence and diversity of the two methods also differ.

B. Convergence and diversity of AIS

Convergence ability:

AIS implements many methods to increase the ability to converge:

- All of AIS's selection, replication and transformation are based on the appropriateness of the elements. When an element has a greater suitability, the probability of being selected is greater, and the replication rate is also greater (less changed).

- Methods of maintaining good memory memorization, saving the good elements of generations, and adding to the population of generations, this also significantly increases the ability of the algorithm to converge. .

- Methods of eliminating elements of low relevance and eliminating identical elements based on the interaction of the solutions with the shortest path value, the interaction of the solutions together increases the quality of the pants as well as increase the possibility of convergence of AIS.

AIS supports the following methods to maintain the diversity of the population:

- Although the operations of selection, replication, and transformation are performed based on the appropriateness of the elements. But the selection and mutation are done randomly.

For example, the selection and selection mechanism: Firstly randomly select a number of elements in the population, determine the suitability of the elements, then select the highest-appropriate element. The selection of a good element is based on the suitability of the element, but the first step of the selection process is still random of some elements.

In mutations, when mutations per bit sequence, although the number of bits will be mutated inversely proportional to the suitability of that string, but the bits selected for mutation are random unknown. This helps maintain the diversity of the population.

- New method of adding new elements: Each generation of AIS creates a new random population to replace bad elements, which helps AIS to maintain diversity.

C. The convergence and diversity of GA

Convergence ability: GA supports the following methods to ensure the ability of convergence

- Selective method: GA selects elements to transform based on the appropriateness of the elements. Elements with higher relevance are more likely to be selected.

- But when performing hybrid and mutation transformations, they are performed randomly without relying on the appropriateness of the elements. The location chosen for hybridization is random, all elements are mutated at the same rate, this rate is fixed from the beginning. So the convergence ability of GA may be slower than AIS.

- GA has the method to remove the best suitable elements, but only based on the interaction of the solution with the shortest value of the path without relying on the interaction of the solutions. of GA, within the GA population there may be very similar elements coexisting.

- GA maintains only the best element, there is no method to maintain the memorization of good elements, this also partly limits the ability of GA to converge.

Diversity of populations:

- In the selection methods of GA as well as the next AIS based on the suitability of the elements it still ensures randomness thanks to the selection mechanism.

- GA transformations are completely not based on the suitability of the elements, this increases GA's diversity significantly.

- GA has a method of removing elements with good suitability, but only based on the interaction of the solution with the shortest path length value without relying on the interaction between the solutions. This limits the quality. The amount of GA population, in GA population, can have very similar elements coexist, making the diversity of GA population somewhat reduced.

- GA does not have a method to generate new populations randomly in each generation. This mechanism also reduces the diversity of the GA population.

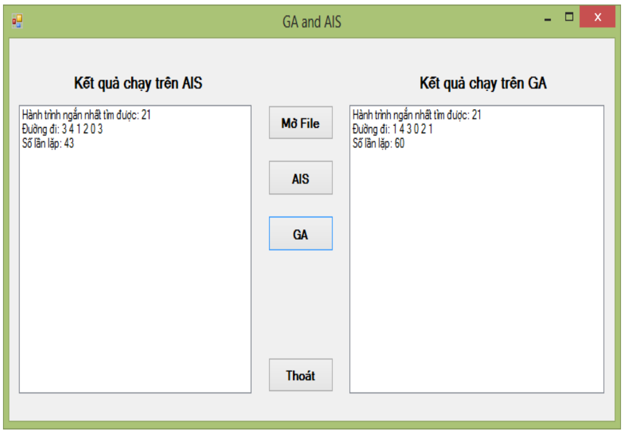
D. Experiments

Each input for experiments include: a number of cities to visit n, and cost of the roads in the form of triples (u, v, c) where c is cost (in length) from city u to city v. Cities are numbered from 0 to n-1.

The 1st experiment with 5 cities:

INPUT	OUTPUT	
	AIS	GA
5	21	21
0 1 5	3 4 1 2 0 3	1 4 3 0 2 1
0 2 3	43	60
0 3 1		
0 4 7		
1 2 4		
1 3 9		
1 4 6		
2 3 11		
2 4 4		
3 4 7		

Some results are in a implementation:



Results from AIS:

Route: 3 4 1 2 0 3
Length of the route: 21
Number of iterations: 43

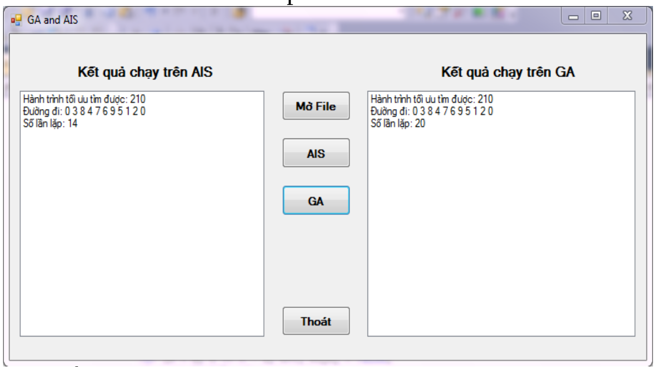
Result from GA:

Route: 1 4 3 0 2 1
Length of the route: 21
Number of iterations: 60

The 2nd experiment with 10 cities:

0 1 65	1 2 12	2 4 79	3 7 65	5 7 84
0 2 28	1 3 76	2 5 92	3 8 27	5 8 91
0 3 46	1 4 12	2 6 81	3 9 10	5 9 16
0 4 90	1 5 21	2 7 75	4 5 46	6 7 31
0 5 64	1 6 58	2 8 5	4 6 60	6 8 58
0 6 90	1 7 42	2 9 66	4 7 10	6 9 11
0 7 74	1 8 76	3 4 87	4 8 8	7 8 58
0 8 62	1 9 67	3 5 66	4 9 38	7 9 78
0 9 77	2 3 76	3 6 69	5 6 75	8 9 89

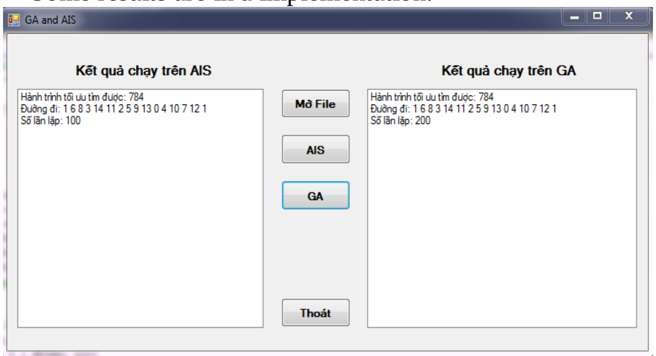
Some results are in a implementation:



The 3rd experiment with 15 cities

0 1 156	1 3 185	2 6 194	3 10 29	5 6 14	6 13 184	9 10 33
0 2 79	1 4 149	2 7 169	3 11 191	5 7 179	6 14 121	9 11 50
0 3 3	1 5 174	2 8 153	3 12 185	5 8 186	7 8 78	9 12 108
0 4 19	1 6 116	2 9 137	3 13 169	5 9 32	7 9 59	9 13 17
0 5 145	1 7 183	2 10 122	3 14 93	5 10 24	7 10 5	9 14 171
0 6 4	1 8 2	2 11 17	4 5 189	5 11 88	7 11 9	10 11 179
0 7 45	1 9 36	2 12 124	4 6 93	5 12 93	7 12 77	10 12 112
0 8 123	1 10 184	2 13 147	4 7 176	5 13 32	7 13 162	10 13 114
0 9 66	1 11 143	2 14 43	4 8 169	5 14 161	7 14 197	10 14 4
0 10 199	1 12 42	3 4 170	4 9 127	6 7 176	8 9 157	11 12 175
0 11 154	1 13 166	3 5 7	4 10 91	6 8 4	8 10 126	11 13 179
0 12 126	1 14 147	3 6 154	4 11 118	6 9 144	8 11 53	11 14 43
0 13 54	2 3 114	3 7 42	4 12 174	6 10 81	8 12 200	12 13 196
0 14 7	2 4 94	3 8 63	4 13 147	6 11 154	8 13 47	12 14 4
1 2 5	2 5 111	3 9 23	4 14 135	6 12 144	8 14 125	13 14 118

Some results are in a implementation:



Experimental results show that CLONALG outperforms GA to solve the TSP in term of time complexity.

IV. CONCLUSIONS

This article has provided an overview of AIS, especially CLONALG and its applications in optimisation. We can see that the diversity of AIS and GA is almost the same, each method has its own strengths and weaknesses in maintaining the diversity of the population. However, the convergence rate of AIS will be better than that of GA. It has many methods to orient the population, as well as improve the quality of the population very well. The test results from all three experiments also confirm this.

In the future, we will develop this research in the following directions: 1-Improve CLONALG to apply in wide range of optimal problems. 2- study the other algorithms of AIS such as: AiNet, RAIN.

REFERENCES

[1] H. Yang, T. Li, X. Hu, F. Wang, Y. Zou, "A survey of artificial immune system based intrusion detection", The Scientific World Journal, 2014.

[2] J. Brownlee, "Clonal Selection Algorithms", Technical Report 070209A, Complex Intelligent Systems Laboratory (CIS), Centre for Information Technology Research (CITR), Faculty of Information and Communication Technologies (ICT), Swinburne University of Technology, 2007.

[3] J. Timmis and C. Edmonds, "A Comment on Opt-AiNet: An Immune Network Algorithm for Optimisation," in Proceedings of the Genetic and Evolutionary Computa-tion (GECCO), 308–317, 2004.

[4] De Castro L. N. and Timmis J., An artificial immune network for multimodal function optimization, in Proceedings of the 2002 Congress on Evolutionary Computation, 699–704, 2002.

[5] K. B. Sim, D. W. Lee, "Modeling of Positive Selection for the Development of a Computer Immune System and a Self-Recognition Algorithm", International Journal of Control, Automation, and Systems, 1(4), 453–458, 2003.

[6] Kelsey, Johnny; Timmis, Jon. "Immune Inspired Somatic Contiguous Hypermutation for Function Optimisation". Genetic and Evolutionary Computation, 2003.

[7] Maoguo Gong, Licheng Jiao, and Lining Zhang, Solving Traveling Salesman Problems by Artificial Immune Response, Asia-Pacific Conference on Simulated Evolution and Learning, 64-71, 2006.

[8] Y. Tan, "Anti-Spam Techniques Based on Artificial Immune System", CRC Press, 2016.

[9] Zhao, Jun and Liu, Quanli and Wang, Wei and Wei, Zhuoqun & Shi, Peng, A parallel immune algorithm for traveling salesman problem and its application on cold rolling scheduling. Inf. Sci.. 181. 1212-1223, 2011

[10] Z. Fuyong, Q. Deyu, "A positive selection algorithm for classification", Journal Computational Information Systems, 207–215, 2012.

BIOGRAPHY

Nguyen Nhu Trang is a lecturer in the Thai Nguyen University of Medicine and Pharmacy. He received a Bachelor of Informatics in 2004. He finished her master course on Computer science at Thai Nguyen University of Information and Communication Technology in 2008. He has published several papers in some Vietnamese journals.

Doan Thi Minh Thai is a lecturer in the Faculty of Mathematics at Thai Nguyen University of Education, from where she received a Bachelor of Informatics in 2004. She finished her master course on Computer science at Thai Nguyen University of Information and Communication Technology in 2007. She teaches Discrete Math, Design and Analysis of Algorithms.