A Comparative Analysis of Immune System Inspired Algorithms for Traveling Salesman Problem

Isha Muthreja and Devinder Kaur
EECS Department
University of Toledo
Toledo, OH, USA 43606
devninder.kaur@utoledo.edu
isha.muthreja@rockets.utoledo.edu

Abstract — *In this paper, Traveling Salesman Problem was implemented using Clonal Selection Algorithm and the modified version of Clonal Selection Algorithm. The performance of both the algorithm was compared with pre-existing approaches GA (Genetic Algorithm) and BB (Branch and Bound) for the same problem. The distance covered in the tour was measured for each algorithm by varying the number of cities. The run time for the algorithm was measured to evaluate the performance of TSP. The two proposed approaches - CSA and its modified version were compared against - GA and BB methods and the results indicate that the proposed methods provide better solution to TSP with respect to the shorter tour distance and shorter time of convergence as compared to the conventional approaches.*

Keywords **— AIS, CSA, TSP, Immuno-computing, GA, CLONALG, BB.**

1 Clonal Selection Algorithm

Natural biological systems are often used as a source of inspiration for solving engineering problems in different areas[4]. CSA is such an immune system based algorithms that have tremendous potential in many engineering applications. In the past few years, several artificial immune models based on the biological immune response network have been proposed because they help us to reach global optimal for real world problems.

CSA takes inspiration from the natural immune system, one of the most intricate biological systems, composed of complex structures and cells. The organs composing the immune system are called as lymphoid organs[1]. The lymphoid organs consist of lymphocytes, which are white blood cells that contains receptors to recognize pathogens. B-lymphocytes and T-lymphocytes are two main types of lymphocytes, namely called as B-cells and T-cells.

The B-cells are capable of fighting against diseases in the bone marrow, whereas the T-cells are responsible for fighting against diseases in the thymus. Both represent several important features from the biological and computational perspectives. They both are capable of recognizing specific molecular patterns present on pathogens and multiplying themselves through a cloning process to fight against them. All blood cells that generate and maturate within the bone marrow become ready to act as an immune cell within the bone marrow, and are called B-cells; whereas, the cells that are generated within the bone marrow but migrate to the thymus and maturate there, are called T-cells. After reaching the thymus, they become immune-competent cells and learn to distinguish between cells of the organism and the invading cells.

The immune system of the human body is capable of recognizing the foreign cells entering a human body as they can be harmful and can cause diseases. These foreign cells are called **Antigens**. In such conditions, our body learns how to neutralize the effects of antigens by understanding the behavior pattern and opposing them to maintain stability of the body. The cells produced by the human body to fight with antigens are called as an **Antibody**. Immune system controls the action of antigen internally. The Clonal Selection Algorithm works on the concept that it filters the antibodies based on affinity. The degree of recognition between antibody and antigen is called as **Affinity**. The better the recognition, the higher the affinity, and vice-versa. Antibodies capable of fighting against antigens are subjected to cloning proportional to affinity. To flourish in the next generation, the clonal-set competes with the antibody population. During the process, the population having higher-affinity replaces the clonal-set having a low-affinity population.

The paper is organized as follows: Section II presents an introduction to Traveling Salesman Problem. It gives a brief description about how TSP works. Section III gives details about Clonal Selection Theory. Section IV describes about the implementation of CSA algorithm. This section focuses on pseudo code and describes the flow of the algorithm. Section V addresses the simulation of TSP using CSA and modified version of CSA. Section VI shows the comparative analysis among four different algorithms of Traveling Salesman Problem. There is a comparison between two proposed methods – CSA and modified CSA with two existing approaches - GA and BB. Finally, Section VII concludes the paper, discusses the application areas and some potential future areas of work are suggested.

2 Traveling Sales Man Problem

The TSP aims to find the minimum distance path starting from the origin city, passing through all of the cities only once and returning to the origin city. The reasons why TSP has been attracting the attention of many researchers and still remains an active area of research are because:

- As it is an NP-Complete problem, the number of ways to solve the problem is the permutation of all the cities in the problem.
- A large number of real-world problems can be modeled by TSP.

Traveling Salesman Problem is a common algorithm used in various concepts of daily life. This algorithm could be used for food delivery services for faster service. Cab services: to search shortest tour path between the source and destination. Pooling system in cab – when there are more than one passengers traveling at the same time, then the algorithm decides which path to be followed, in order to pick and drop all the passengers: to minimize the traveling path and time.

Mathematical Explanation for TSP

The salesman starts from a city, visits all other cities only one at a time and finally returns to the original city in such a way that no cities are visited more than once. Given n cities, named C_1, C_2 ……. Cn, the objective is to choose the path such that the sum of all distances between each city and its successor is minimized. The successor of the last city in the permutation is the first one. The distance d, between any two cities with coordinates (x_1, y_1) and (x_2, y_2) is calculated by [7] the simple distance formula given as Formula (1):

$$\text{Distance} = \text{sqrt}\left((x_1-x_2)^{\wedge 2} + (y_1-y_2)^{\wedge 2}\right) \qquad (1)$$

3 Clonal Selection Theory

M. Burnet and D. Talmage[11], who in the mid-1900s proposed and formalized the Clonal Selection Theory of adaptive immunity (Burnet, 1959), broadly accepted as an explanation of how the adaptive immune system responds to pathogens. Clonal Selection Theory is the basis of modern immunology. According to the Clonal Selection Theory, the immune system consists of sets of discrete cells and molecules that remain undisturbed unless a pathogenic agent invades the organism. After the invasion, some subset of the immune cells recognizes the invading antigen and binds with it. This process is done by either B-cells or T-cells, which stimulate the immune cells capable of recognizing the antigen to start reproducing itself. Cellular reproduction is asexual and occurs through the process of cell division. This phenomenon is also called as clonal expansion, which generates clone cells. These clone cells subject to an error while forming new antibodies, which causes change in the structure of cell. These clone cells are capable of recognizing a specific type of antigen and the process is called as mutation.

The main features of the Clonal Selection Theory [2] are:

- Generation of a new random population of diverse antibodies by a process of accelerated mutation.
- Removal of antibodies with affinity less than antigen and retention of all the antibodies with affinity greater than antigen.
- Cloning of antibodies and differentiation on contact with antigens.

Figure 1 shows the Clonal Selection and affinity maturation processes. From the repertoire of B-cells, the one presenting higher affinity with the antigen are selected and stimulated to proliferate. During each proliferation stage, mutation may occur, thus allowing the cell receptor to become more adapted to the antigen presented[3]. Those mutated immune cells with higher affinity with the antigen are selected to become memory cells and antibody secretor (plasma) cells[10].

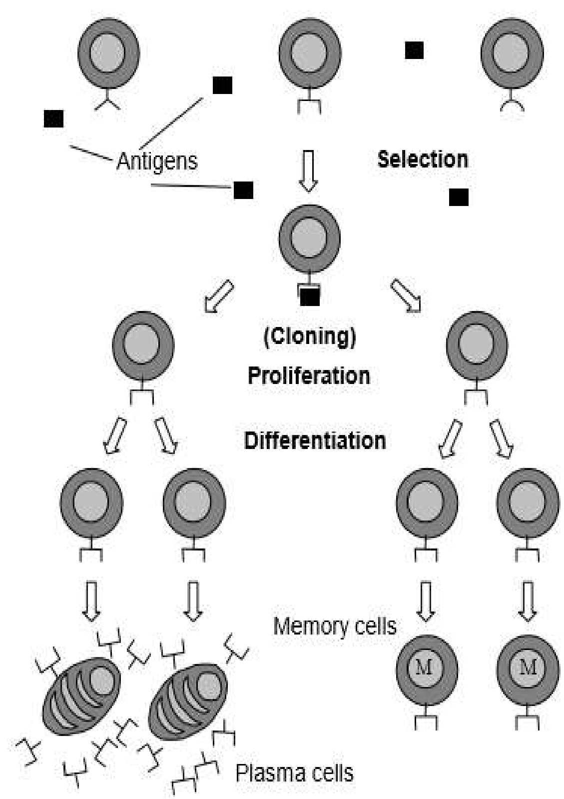

Fig.1. The Clonal Selection Principle, taken from [5].

4 Implementation of Algorithm

In order to develop computational model based on CSA for the TSP, the terminologies[8] of immune system need to be mapped to the structure of TSP. Table I shows the interpretation of CSA terminologies to TSP.

TABLE 1: Mapping of immune system based on TSP

Immune System	**TSP Model**
Antigen (Ag)	Initial Tour Path
Antibody (Ab)	Feasible Tour Path
Gene	A city represented by x and y coordinate
Affinity	Difference of Tour distance between Antigen and Antibody
Cloning	Creation of multiple copies of Antibody
Mutation	Change in one or more gene of Antibody
Population	Number of Antibodies
Generation	Number of Iteration

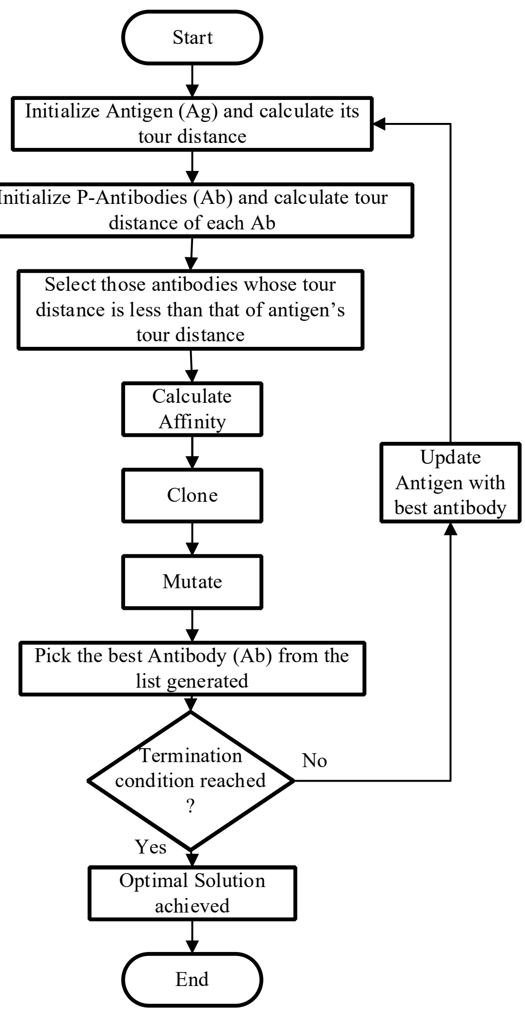

Fig.2. Flow chart diagram, describing stages of the CSA.

The workflow diagram of the TSP based on CSA is shown in Figure 2. It consists of the following steps.

4.1 Initialization of Antigen

In immunology, an antigen refers to the foreign cell that attacks the human body such as virus, parasite, fungus, etc. In the context of TSP, an antigen is a potential solution based on which the final optimum solution is evaluated. This potential solution will help to select those antibodies in each generation whose tour distance is less than the tour distance of antigen. First a random antigen is created which represents the possible tour path as shown in Figure 3. There are as many cells in the antigen as the number of cities to be traveled. Each cell contains the x-coordinate and y-coordinate of the city. Figure 3 represents the sequence of cities to be followed by the salesman.

48,87	5,41	50,36	87,78	23,55 …
…96,24	70,46	10,75	94,73	73,28

Fig.3. A Sample Antigen

4.2 Initialization of Antibody

In immunology, an antibody refers to the cells within the human body such as T-Cells and B-Cells, which fight, with the foreign cells that attacks it. As in immune system, in response to antigen, several antibodies are created to fight it. In simulated CSA for TSP, the antibodies are first created randomly, within the constraints of the problem, i.e., every city must be visited once and only once. The algorithm generates P-number of such antibodies having different tour path sequences. For this algorithm, 50 such antibodies are created.

4.3 Tour Distance Calculation

The distance traveled by salesman to complete the tour is calculated using distance matrix. Tour distance is evaluated for antigen and all the antibodies, which have been initialized in previous step. These tour distances depends on the sequence of path represented in each antibody.

4.4 Selection of Antibodies

The tour distance represented by each antibody is computed and if it larger than the tour distance represented by antigen, it is removed. If the tour distance of antibody is less than antigen then that antibody is selected.

4.5 Affinity Calculation

For the selected antibodies, the affinity is calculated which is defined as the difference between tour distance of antigen and the tour distance of antibody, given in Formula (2). Antibodies having higher affinity when compared with antigen are considered better. The greater the antibody-antigen affinity, better the chances of obtaining optimum solution.

$$Affinity = Tour\ distance_{Antigen} - Tour\ distance_{Antibody} \qquad (2)$$

4.6 Clone the Antibodies

Cloning operation is to make copies of the antibodies.

- Cloning for CSA

For CSA algorithm, the number of clones N_c is fixed to ten and is same for all selected antibodies. Therefore, for every iteration 10 clones of each antibodies are created.

- Cloning for modified CSA

For modified CSA algorithm, the number of clones N_c for each antibody is given by Formula (3) which represents normalization in the range of 0 to 10. N_i is fixed to higher bound value of normalization. In this paper, maximum number of clones generated is 10 hence, N_i is 10. Normalization depends on the x_{max} and x_{min} value where x_{max} is the highest affinity value and x_{min} is the lowest affinity value from the list of antibodies selected in particular iteration. x_i is the affinity value of i^{th} antibody.

$$N_c = ceil\left(N_i \frac{x_i - x_{min}}{x_{max} - x_{min}}\right) \qquad (3)$$

Where N_c is the number of clones to be generated. Ceil (.) is the operator that rounds its argument towards the next higher integer. The higher the affinity, greater the number of clones generated and vice-versa.

TABLE II: Tour distance, affinity and number of clones for CSA and mCSA.

	Tour distance	Affinity	Number of Clones	
			CSA	m _CSA
Antibody 1	45	18	10	4
Antibody 2	23	40	10	9
Antibody 3	19	44	10	10
Antibody 4	58	5	10	0
Antibody 5	30	33	10	8

The tour distance, affinity and number of clones generated for both – CSA and mCSA algorithms are shown in Table II.

4.7 Mutation

During the cloning of antibodies, they undergo mutation while forming new antibodies. Mutation changes the structure of the cells so that they can better recognize and fight against antigen. Mutation operation of immune system is an important step as it enhances the local search mechanism and helps to reach the global solution faster. There are many types of mutation operations like shift change, inversion, transpositions, etc. In this paper, reciprocal exchange type of mutation is used.

- Mutation for CSA

For CSA algorithm, a single point reciprocal exchange mutation operation occurs in which two random points are picked to exchange gene with each other within an antibody. Figure 4 represents a sample mutation process.

- Mutation for Modified CSA

For modified CSA algorithm, each of the cloned population undergoes the hyper mutation scheme [9]. In hyper mutation, the number of mutation points is high when the affinity is low and vice-versa. This process is called affinity maturation [2]. Figure 5 represents a sample reciprocal exchange mutation process. For modified CSA algorithm, reciprocal exchange mutation operation, the number of mutation points is calculated based on Formula (4).

$$M_p = 2 * ((N_i+1)-N_c) \qquad (4)$$

Where M_p is a number of mutation points and N_c is the number of clone and N_i is the higher bound of normalization.

48,87	5,41	50,36	87,78	23,55	96,24	70,46	10,75	94,73	73,28

a) Original Antibody (Sequence :1,2,3,4,5,6,7,8,9,10)

48,87	70,46	50,36	87,78	23,55	96,24	5,41	10,75	94,73	73,28

b) Antibody after Mutation (Sequence :1,7,3,4,5,6,2,8,9,10)

Fig.4. Reciprocal exchange Mutation for CSA algorithm

48,87	5,41	50,36	87,78	23,55	96,24	70,46	10,75	94,73	73,28

a) Original Antibody (Sequence :1,2,3,4,5,6,7,8,9,10)

48,87	70,46	23,55	87,78	50,36	96,24	5,41	73,28	94,73	10,75

b) Antibody after Mutation (Sequence :1,7,5,4,3,6,2,10,9,8)

Fig.5. Reciprocal exchange Mutation for modified CSA algorithm.

4.8 Stopping Criteria

The algorithm is stopped when termination condition are reached; which is either the desired tour distance has been found or the set number of iterations are achieved.

Algorithm 1: Pseudo Code for CS algorithm.

```
procedure [P] = CLONALG (max)
    initialize P                    //P-Antibodies
    t ← 1
    initialize Ag                   //initialize Antigen
    while t <= max do,              //for max iterations
     f  ← eval (P)                  //determine the distance
     P1 ← select (P, f)   //Select antibodies having distance higher than distance of antigen
     A  ← calc (Affinity)           // Calculate Affinity
     C  ← clone  (P , A)            //proportional to affinity
     C1 ← mutate (C , A)            //inversely proportional
     Ab ← select (C1, A)            //select the best cloned Ab
     Ag ← update (Ab)               //update Antigen with Antibody
     t  ← t+ 1
    end while
end procedure
```

As per the pseudo code for CSA, the algorithm has the following steps to achieve optimal solution:

1. **Initialization of Antigen** – Randomly select an antigen. The structure of an antigen represents the path of the tour or sequence of genes. Each gene represents coordinates of city in two-dimensional graph.
2. **Initialization of Antibodies** –Generate a set of antibodies (P), which are antibody population for initial generation..
3. **Calculation of Tour Distance** – Calculate the tour distance of Antigen (Ag) and P-Antibodies (Ab).
4. **Selection** – Select all antibodies with tour distance less than the tour distance represented by Antigen.
5. **Affinity Calculation** – Affinity is the difference between tour distance represented by antigen and tour distance represented by antibody.
6. **Cloning** – Cloning is to make copies of the selected antibodies.
7. **Mutation** – Mutate all these copies by reciprocal exchange method of swapping two random genes with each other.
8. **Repeat** – Update the antigen for next generation, with the selected antibody having minimum tour distance from the current generation. Repeat above steps 2 to 7 until the termination criteria is achieved.
9. **Termination Condition** – Termination criteria is either the desired tour distance has been found or the set numbers of iteration is reached.

5 Simulation on TSP

The Traveling Salesman Problem using two approaches – CSA and modified CSA was implemented to compare the performance. The results obtained are presented in this section. The number of cities are varied in each run and results are observed.

6 Clonal Selection Algorithm

TABLE III. Results of the CSA algorithm for varying number of cities

Cities	Tour Distance	Convergence Time (sec)
10	2.85E+02	79.99
20	7.85E+02	81.00
30	1.16E+03	81.69
40	1.71E+03	83.22
50	2.14E+03	86.83

Table III provides us tour distance for 10, 20, 30 , 40 and 50 cites respectively. It also provides the convergerence time taken (in seconds) of the algorithm to achieve the minimum tour distance.

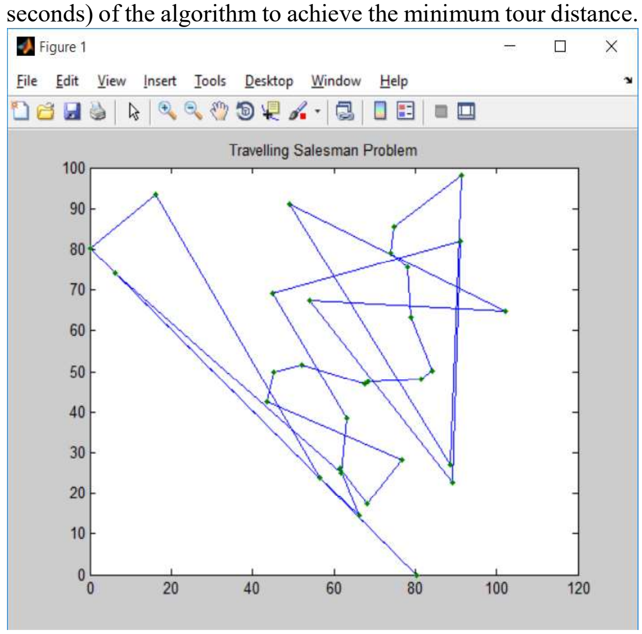

Fig.6. Graph showing best path for 30 cities TSP using CSA.

Figure 6 shows the best solution of TSP using CSA for 30 cities to determine the minimum tour distance. The optimum path of tour distance was taken when no further improvement of the tour distance was observed.

7 Modified Clonal Selection Algorithm

Table IV provides the results for TSP using modified CSA. The tour distance is the optimum path covered by the salesman and convergence time is the time taken (in seconds) by the algorithm to find the optimum path.

TABLE IV. Results of modified CSA algorithm for varying number of cities

Cities	Tour Distance	Convergence Time (sec)
10	2.84E+02	8.36
20	7.50E+02	15.13
30	1.11E+03	17.97
40	1.70E+03	18.08
50	2.08E+03	29.30

Figure 7 shows the graph for best solution of TSP using modified CSA for 30 cities to determine the tour distance.

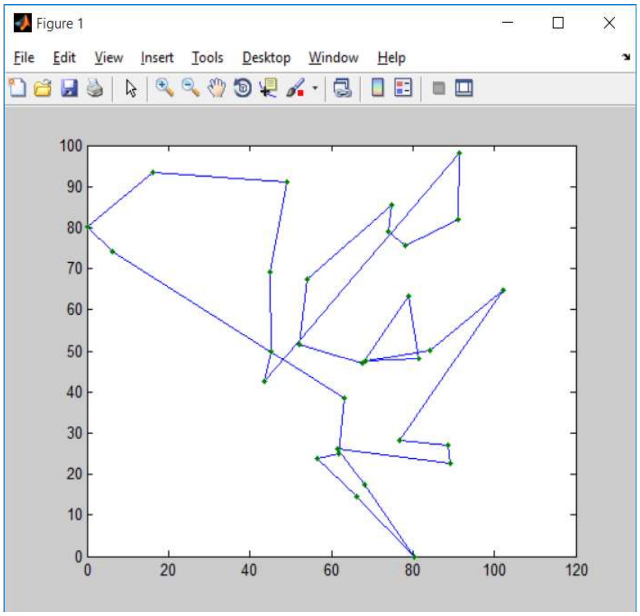

Fig.7. Graph showing best path for 30 cities TSP using modified CSA.

8 Comparison

This study compares four algorithms for the TSP: Standard TSP using branch and bound approach, TSP using GA[7], TSP using CSA and TSP using modified CSA. The coordinates of the city were taken in the input file and distance between each city is generated through distance matrix using coordinates from the input file for all the four algorithms.

TABLE V. Comparison of Optimum distance for the tour each algorithm having a varying number of cities

Cities	BB	GA	CSA	mCSA
30	1.24E+04	6.85E+03	1.16E+03	1.11E+03
50	1.19E+05	1.78E+04	2.14E+03	2.08E+03
70	5.20E+04	1.27E+04	3.15E+03	3.03E+03
90	2.47E+05	1.77E+04	4.06E+03	4.03E+03

Table V shows the optimum distance for each algorithm. The tour distance traveled in branch and bound method is relatively high when compared to other algorithms. For GA, the tour distance was comparatively less than that of branch and bound method. For CSA and mCSA, the optimum distance was relatively less than other existing approaches. However, modified CSA shows the best results having the optimum tour distance less than all other algorithms.

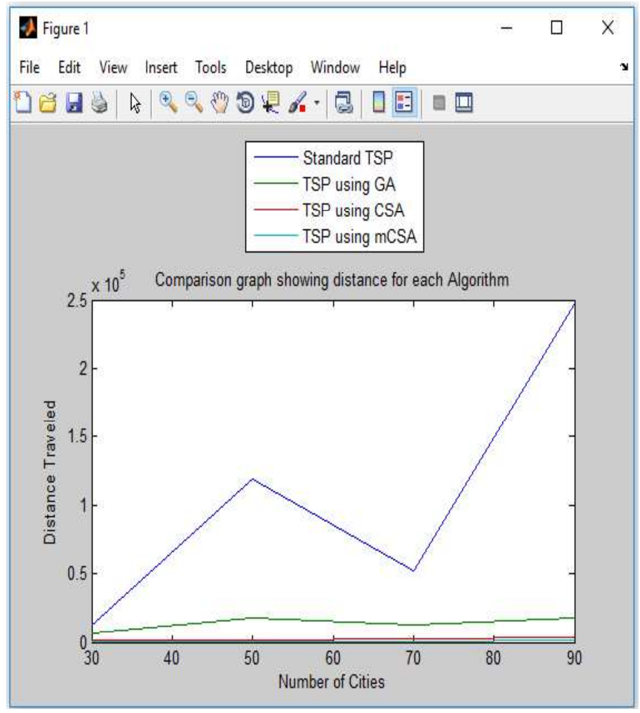

Fig.8. Graph shows comparison of Distance Traveled and Number of cities for four different algorithms.

Figure 8 shows the graph that compares the tour distance of the four algorithms for cities ranging from 30 to 90. The optimum path is taken when there is no further improvement in each algorithm.

TABLE VI. Comparison of convergence time to achieve optimum tour for each algorithm having varying number of cities

Number of Cities	Convergence Time (sec)			
	BB	GA	CSA	mCSA
30	700	127.35	81.69	17.97
50	630	209.99	86.83	29.30
70	620	319.89	100.11	32.12
90	740	394.24	120.42	18.31

Table VI shows the observation for convergence time to obtain minimum tour distance by the four algorithms – Branch and bound, GA, CSA and modified CSA. Criteria for convergence was chosen when no further improvement in tour distance was observed for two successive iterations. The branch and bound method takes the maximum time and is slower when compared to other three algorithms. TSP using GA takes a long time for searching the optimum tour path. TSP using CSA takes less time compared to standard TSP and TSP using GA.

However, it was observed that modified CSA takes the minimum time to converge.

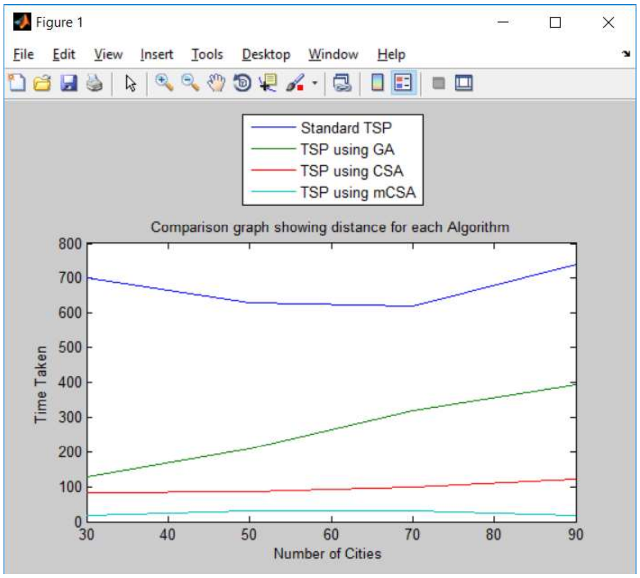

Fig.9. Graph shows comparison of Convergence Time vs. number of cities for four algorithms.

Figure 9 shows the graph that compares the time taken for the convergence of tour by the four algorithms for cities ranging from 30 to 90. It can be inferred the mCSA algorithm converges faster than all other algorithms with the increase in the number of cities. Therefore, it is observed that the convergence rate, the quality mCSA is better than CSA algorithm.

9 Conclusion

In this paper, four Traveling Salesman Problem are compared, with an aim to achieve minimum tour distance. The four TSP are Standard TSP using branch and bound, TSP using GA, TSP using CSA and TSP using modified CSA. Based on the results of the tour distance obtained by each algorithm, it has been found that modified Clonal Selection Algorithm is the best evolutionary approach.

Clonal Selection Algorithm inspired from Artificial Immune System uses cloning and mutation strategies to solve TSP. The algorithm makes an improvement in achieving minimum tour distance by the introduction of reciprocal exchange mutation technique. The experiment results shows that the modified Clonal Selection Algorithm is better than other evolutionary algorithms for Traveling Salesman Problem and it can achieve minimum tour distance in a reasonable good convergence time.

The application of the CSA is mainly to perform learning and memory acquisition. It can be applied in many other areas like pattern recognition problem; to illustrate the application of various artificial neural networks; in areas of multimodal optimization [6] task to evaluate its capability of locating and maintaining a stable population of multiple optima solutions. They can also be applied in other numerous application, from VLSI circuit design, to games like Pokémon Go; in food delivery services and cab services.

ACKNOWLEDGMENT

Isha Muthreja would like to thanks Dr. Devinder Kaur for her suggestion and advice, without her guidance and support I would not have completed the project.

REFERENCES

[1] L. N. de Castro, F. J. Von Zuben, “Immune and neural network models: theoretical and empirical comparisons”, International Journal of computational Intelligence and Applications, vol. 1, pp. 239-257, 2001.

[2] L. N. de Castro, F. J. Von Zuben, “The Clonal Selection Algorithm with engineering applications”, Workshop Proceedings of GECCO, pp. 36-37, 2000.

[3] Shangee GAO, Hongwei Dai, Gang Yang and Zheng Tang, “A Novel Clonal Selection Algorithm and Its Application to Traveling Salesman Problem”, Vol.E90, No.10, Oct 2007.

[4] Yilmaz Atay, Halife Kodaz, “Optimization of job shop scheduling problems using modified Clonal Selection Algorithm”, doi: 10.3906/elk-1212-26, 22: 1529-1539, 2014.

[5] Leandro Nunes de Castro, “Fundamentals of Natural Computing”, 267–323, edition 2007 by Taylor & Francis Group.

[6] L. N. de Castro, F. J. Von Zuben, “Learning and Optimization using the Clonal Selection Principle”, IEEE Transactions on evolutionary Computation, special issue on Artificial Immune Systems, vol. 5,pp 291-317 , 2002.

[7] Vanni Wadhawan, Seema Rani, “Optimization of TSP using Genetic algorithm”, Vol. 2, Ver. 3, IJArcsT 2014.

[8] Yaping Jiang and Yalin Wang, "A Method for TSP Immune-Based", Vol. 1, No.4, 2012.

[9] Ayi Purbasari, Iping Supriana Suwardi, Oerip Santoso, and Rila Mandala, "Computer Experiments with a Clonal Selection Algorithm”, pp 59-71, Vol. 10, 2014.

[10] Berna haktanirlar Ulutas and Sadan Kulturel Konal, “A review of Clonal Selection Algorithm and its application”, DOI 10.1007/s10462, January, 2011.

[11] Jason Brownlee, “A review of Clonal Selection Theory of Acquired Immunity”, CIS Technical report 070223A, February 2007.