Optimization of Path Finding Algorithm using Clonal Selection: Application to Traveling Salesperson Problem

Tejbanta Singh Chingtham, *Member IEEE, IACSIT and IAENG,* G. Sahoo, MIAENG and M. K. Ghose

Abstract— **The Clonal selection is a mechanism used by the natural immune system to select cells that recognize the antigens to proliferate. The proliferated cells are subject to an affinity maturation process, which improves their affinity to the selective antigens. The concept of Clonal selection is an important one to the success of the human immune system, and it provides an excellent example of the principles of selection at work. The positive and negative selection is another interesting mechanism in the immune system that works together to both retain cells that recognize the self peptides, while also removing cells that do not recognize any self peptides. In this paper, a cloning-based algorithm inspired by the Clonal and the positive/negative selection mechanism of the natural immune system is presented. The well known TSP is used to illustrate the approach. Simulations demonstrate that this approach generates good solutions to traveling salesman problem.**

Index Terms—**Artificial Immune System, Clonal Selection, Hamiltionian circuit, Negative/Positive selection, Traveling salesman problem.**

I. Introduction

Optimization, a key topic in the areas of engineering and science, is referred to a process of finding the best solution in the most effective way to a given problem, eventually with some constraints. Most known optimization problems like a Traveling Salesman Problem (TSP) have been shown to be NP-hard. Approaches proposed in the literature to solve the NP-hard problems have been divided into two classes: exact approaches and heuristic approaches [3],[15]. Both approaches have their specific properties, advantages, and disadvantages. Exact approaches give exact solution to the studied problem, but they work reasonably fast only for relatively small problem sizes. These exact approaches are branch-and-bound, dynamic programming, and minimum spanning tree approach, but Heuristic approaches deliver either apparently or probably a good solution, but which could not be proved to be optimal [17].

In this paper, an algorithm based on the Clonal selection with a negative/positive selection mechanism is used to solve the problem. By cloning action, an agent do not need to choose between two or more paths, but it clones itself and its clone moves to neighboring node selected at random. Moreover, the number of search agents is not constant and changes during the course of the algorithm due to cloning/suppression operations.

II. Natural Immune System: An Overview

The immune system defends the body against harmful diseases and infections. It is capable of recognizing most antigens' attacks by some certain important immune cells, called B-cells. B-cells circulate through the blood and lymphatic network waiting to encounter antigens (the foreign molecules belonging to pathogens that invade the body). Each antigen has a particular shape that is recognized by the receptors present on the B-cell surface[1],[2]. More precisely, B-cells synthesize and carry on their surfaces molecules, called antibodies that act like detectors to identify antigens. Thus the quality of the antibody is crucial for the immune system to successfully recognize the antigen. If a B-cell is useful to recognize the antigen, it may be stimulated to clone (i.e., proliferate or clonally expand). More precisely, a B-cell with better fitting receptors and binding more tightly the antigen, replicate more and survive longer[11],[3].

To do this, immune system must perform pattern recognition tasks to distinguish molecules and cells of the body (called "self") from foreign ones (called "nonself") [12],[7]. Thus, the problem that the immune system faces is that of distinguishing self from dangerous nonself. These foreign proteins (kinds of molecules) must be distinguished from estimated different proteins of self, so recognition must be highly specific. The architecture of the immune system is multilayered, with defenses provided at many levels [4]. The outermost layer, the skin, is the first barrier to infection. A second barrier is physiological, where conditions such as pH and temperature provide inappropriate living conditions for some foreign organisms (pathogens). Once pathogens have entered the body, they are handled by the innate immune system and by the adaptive immune response [5],[6].

The innate immune system consists primarily of circulating scavenger cells such as macrophages that ingest extra cellular molecules and materials, clearing the system of both debris and pathogens. The adaptive immune response (also called

Manuscript received August 27, 2009. This work was supported in part by the AICTE RPS under Grant 8023/BOR/RID/RPS-235/2008-09.

Tejbanta Singh Chingtham is with the Sikkim Manipal Institute of Technology, Majitar, Rangpo, East Sikkim – 737132, INDIA as a Reader in the Department of Computer Science and Engineering. (e-mail: cts@smuhmts.edu; phone +919734916135).

G. Sahoo is with Birla Institute of Technology, Mesra, Ranchi, Jharkhand as a Professor in Department of Information Technology. (e-mail: gsahoo@bitmesra.ac.in)

M. K. Ghose is with Sikkim Manipal Institute of Technology, Majitar, Rangpo, E-Sikkim, as a Professor in Department of Computer Science and Engineering (mkghose@smuhmts.edu)

"the acquired immune response") is the most sophisticated and involves many different types of cells and molecules. It is called "adaptive" because it is responsible for immunity that is adaptively acquired during the lifetime of the organism. Because the adaptive immune system provides the most potential from a computer security viewpoint, we will focus on it in this overview. The adaptive immune system can be viewed as a distributed detection system which consists primarily of white blood cells, called lymphocytes.

Lymphocytes function as small independent detectors that circulate through the body in the blood and lymph systems. Lymphocytes can be viewed as negative detectors, because they detect nonself patterns, and ignore self patterns. Detection, or recognition, of nonself occurs when molecular bonds are formed between a pathogen and receptors that cover the surface of the lymphocyte. The more complementary the molecular shape and electrostatic surface charge between pathogen and lymphocyte receptor, the stronger the bond (or the higher the affinity). Detection is approximate; hence, a lymphocyte will bind with several different kinds of (structurally related) pathogens. The ability to detect most pathogens requires a huge diversity of lymphocyte receptors. This diversity is partly achieved by generating lymphocyte receptors through a genetic process that introduces a huge amount of randomness [8].

Generating receptors randomly could result in lymphocytes that detect self instead of nonself, which would then likely cause autoimmune problems in which the immune system attacks the body. Autoimmune disorders are rare because lymphocytes are self-tolerant, i.e. they do not recognize self.

III. The Clonal Selection Approach

The immune response represents solutions and antigens represent the problem to solve. More precisely, B-cells are considered as artificial agents that roam around and explore an environment. The optimization problem represents the pathogen. In other words, the optimization problem is described by an environment of antigens. The positive and negative selection mechanism is used to control the agent proliferation by eliminating useless or bad solutions [12]. Hence, the positive/negative selection rules can be considered as "a reinforcement learning mechanism" that not only selects suitable solutions, but also regulates the agent population size that growth due to the cloning operation [2],[13].

In the immune system the number of cells directed against an antigen increases by proliferation operation when this antigen is present in the body and reduces when it is eliminated. During this operation, a cell changes its morphology such as change of the life duration. So, the proliferation increases the number of agents that improve the affinity with the antigen in order to inhibit and destroy it. In other words, the proliferation corresponds to the creation of new agents. The new created agents are structurally and behaviorally close to their creators but not exactly the same to allow the adaptation of the system. The apoptosis corresponds to the programmed cellular death. This mechanism occurs when a cell is not adapted to the antigen elimination. Thus, useless cells are destroyed [9],[14].

Using the immune-based collective behavior, a clonal and positive/negative selection, the population size of agents in the system is regulated dynamically in order to search the optimal solution to a given problem. In fact, an agent which is estimated unsuitable can be destroyed before being proliferated. The decision is made locally on the agent level; no global controller is necessary.

IV. Problem Description

The problem can simply be stated as follows: the traveling salesman must visit every city exactly once and then return to the starting city. More precisely, the TSP is the problem of finding a shortest tour which visits all cities. Formally, let's consider a graph G=(N,E), where N is a set of nodes representing cities and E is a set of arcs connecting these nodes. The distance between the city i and the city j is denoted. Therefore, a TSP problem consists of finding a minimal length Hamiltonian circuit in the graph G. An Hamiltonian circuit of graph G is a closed tour visiting once and only once all the n = |N| nodes of G, and its length is given by the sum of the lengths dij of all arcs (i,j) that it is composed [3] [15].

V. Mapping between Immune System And TSP Problem

The environment is the city graph wherein nodes represent antigens. B-cells are agents that progress from a city to neighboring cities and can clone or destroy themselves based on positive/negative selection criteria as shown in fig1. The algorithm starts with an initial agent at the source city. At each algorithm cycle, an agent could clone itself and the newly spawned clone moves to neighboring cities[17]. When an agent reaches a city that belongs to its already visited cities set, the positive selection rule is triggered and the agent kills it (i.e., useless solution). Otherwise, the agent clones it and the clone acquires a copy of the already visited cities set from its parent. When all survival agents have accomplished their tour (i.e., reach the source city), the negative selection rule is triggered and among these B-cell agents that constitute the immune response, the one that held the best tour is selected (i.e., useless agents are destroyed) [10],[12].

Immune system	Optimization problem
Pathogen	Problem (environment of antigens) (e.g., city graph wherein nodes represent antigens)
Immune response	Solution (e.g., shortest path)
B-cells	Agents
Clonal selection	Creating new agents in order to explore the environment (i.e.,

	proliferation)
Positive/negative Selection	Selection of useless/bad agents to kill themselves (i.e., apoptosis)

Fig 1. Mapping between Immune system and TSP problem

In order to stop the cloning operation, a better way is to try and develop a tight lower bound of the optimal solution. This lower bound will then act as a stop sign for the search terminating an agent once the upper bound of the partial solution being investigated overlaps the lower bound of the optimal solution. This significantly reduces the useless tours to be explored and consequently reduces the agent's population size. In our case, an agent is allowed to continue to travel if the distance carried is less than to the starting tour generated initially. The nodes are selected randomly by an agent and have an impact to regulate the agent's population. In other words, when an agent reaches a node, the agent clones itself. During its travel, an agent carries the list of visited cities. An agent carried a larger distance kill itself at any city as soon as this condition is detected. This condition ignores a particular path of the TSP graph as soon as it becomes impossible for the path to get a better solution [17],[18].

The positive selection is applied if an agent cannot build a tour or its affinity becomes greater than the affinity of a tour generated initially at random. Also, the negative selection is applied if all suitable agents build a tour (i.e., a feasible solution to the TSP) [19]. In this case, only the agent having a smaller affinity will have remained and is considered the most suitable solution [16].

VI. THE CLONALTSP ALGORITHM

```
// intialiaztion

// define the structure Agent

Agent {

        visited          //list cities visited by the Agent
        notVisted        //list cities still to be visited
        currentNode      // the current node at which Agent
                            is currently in
        pathCost         // path cost of agent from the intial
                            node to the current    node
        complete         // true if the agent returns back to the
                          startnode visiting all node, false
                              otherwise
}
function clonalTSP returns agent

// intialize first Agent
// maxcost is the pre-estimated cost of the tour

Agent.currentNode=startNode
Agent.visited= {NULL}             //initially startNode
Agent.notVisited= {List of all cities}
Agent.pathCost=0                  //Intial pathcost
add Agent to AgentList

while AgentList is not empty

        for each agent in AgentList do

                If Agent.notvisited= {NULL}
            if distance (Agent.currentNode,startNode) > 0
Agent.pathCost=Agent.pathCost+distance(Agent.currentNo
de,startNode)
                Agent.currentNode=startNode
                Agent.complete=true
            Add Agent to AgentCompleteList
            remove Agent from AgentList
                 Agent.visited=Agent.visited U {currentNode}

    remove currentNode from Agent.notVisited

Agent.pathCost=Agent.pathCost+distance(Agent.currentNo
de,A)
                if Agent.pathCost > MaxCost
                remove Agent from AgentList

  if the currentNode has successors still to be visted by the
agent
      select a successor city A of currentNode at random
yet not visited
        Agent.currentNode=A
  if Agent has more than one successor
        ClonedAgent=Agent.clone
Select a successor city B of currentNode other than A at
random yet not visited
        ClonedAgent.currentNode=B
AgentList=AgentList U {ClonedAgent}

    else
      remove Agent from AgentList

    end if

  end for

end while

for each Agent in completeList do
        if minPathCost > Agent.pathCost
            minPathCost=Agent.pathCost
        SaveAgent=Agent
  end if
end for
return SaveAgent
```

VII. COST ANALYSIS: CLONAL SELECTION ON TSP PROBLEM

Case I : 4 Node Network

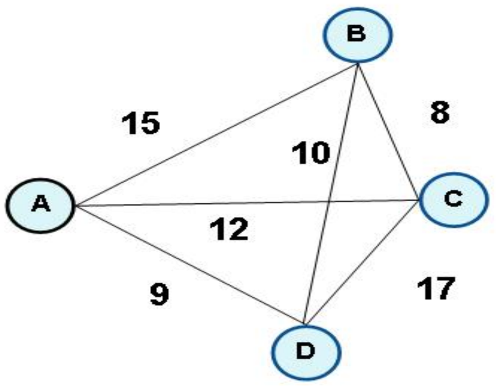

Fig 2. A Sample Path

Maximum Estimated Cost: 54

Total nodes: 4

Maximum Cost path Sequence:

A-B-D-C-A = (15+10+17+12=54)

On using the proposed algorithm the minimum path cost is obtained and is shown in darkened lines in Fig 3.

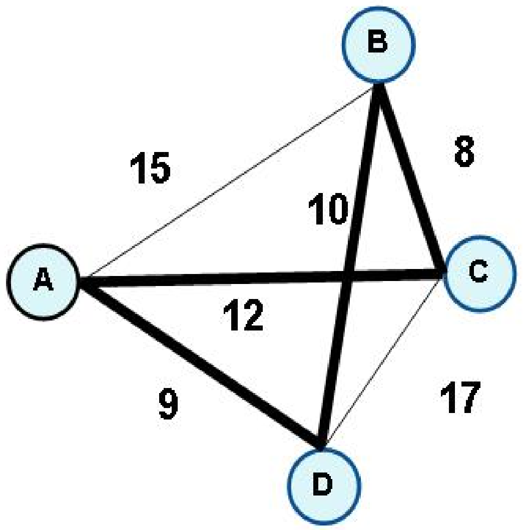

Fig 3. Minimum Cost Optimized Path

Path Cost: 39

Number of Agents generated: 4

Number of completed Agents: 02

Path Sequence: A-C-B-D-A

=12+8+10+9 = 39

Case II: Larger Node Network

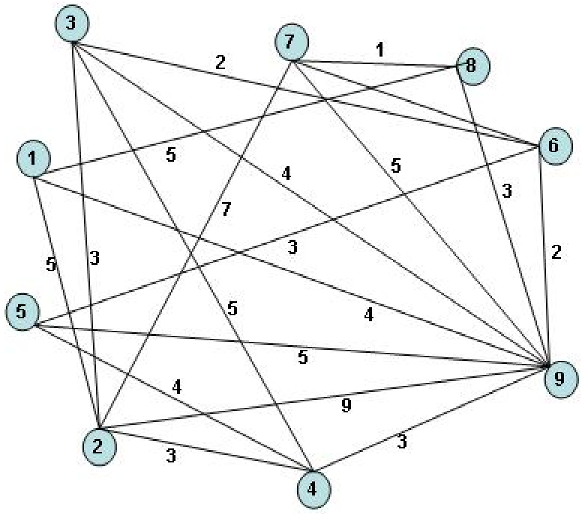

Fig 4. A sample input path for 9 nodes Network

There are various possible feasible solutions for the given problem as there exists many Hamiltonian cycles in the Fig 4. Two possible outputs are:

a) Output 1

Initial Node: 09

Number of Agents generated: 38

Number of completed Agents: 02

Path Sequence: 9-8-7-6-5-4-3-2-1

Path Cost : 30

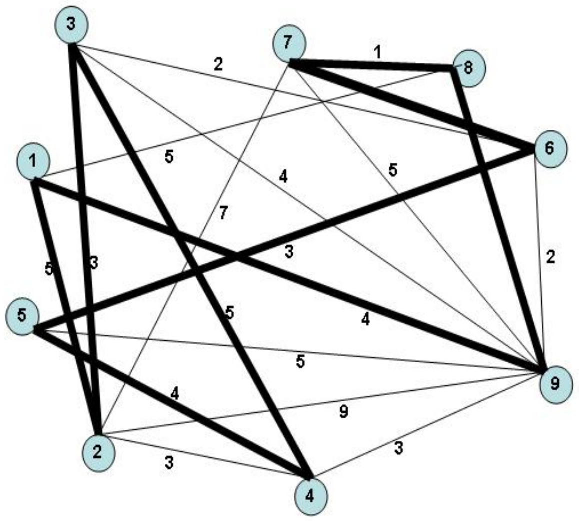

Fig 5. Sample output for input in Fig 4.

b) Output 2

Path Cost: 32

Number of Agents generated: 32

Number of completed Agents: 03

Path Sequence: 5-9-7-8-1-2-4-3-6

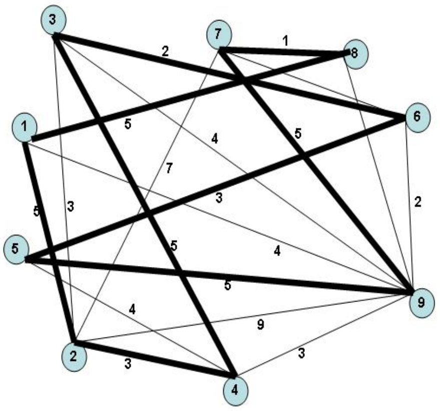

Fig 6. Sample Output Chosen by using Clonal Selection

Optimal Solution

Optimal Path Cost: 30

Path Sequence: 1-2-3-4-5-6-7-8-9

VIII. Result And Conclusion

The other algorithms for the TSP problem are devised using dynamic programming approach and minimal spanning tree algorithms. The minimal spanning tree algorithm is not complete and dynamic programming approach applied along with branch and bound gives complete and optimal solution to the TSP problem with time complexity O(n22n) and space complexity as O(n2n).In the above algorithm, if we consider the time to traverse a node as unit time then time complexity is total number of nodes traversed by each agent. Considering the worst case scenario, first agent generated will traverse all n nodes and subsequent agents traverse n-1 nodes and so on. So, total number of nodes traversed will be 1*n + 1*(n-1) + 2*(n-2) +………. + 2n -2*1 which gives the time complexity of O(n2n). Moreover in average case most of agents will die because of exceeding the maxcost or if agent has no choice to move further .Hence, we can say that the algorithm gives the solution in time less than dynamic programming approach and minimal spanning tree approach.

The algorithm is also complete if there exists a Hamiltonian circuit in the graph but the algorithm does not always results in the optimal solution. Solution obtained from the algorithm is close to optimal one but might not be always optimal [18],[19]. Hence, we can conclude that this algorithm gives fairly good solution in time less than dynamic programming approach.

Acknowledgment

All works described herein is supported under the AICTE, RPS grant vide no. 8023/BOR/RID/RPS-235/2008-09.

References

[1] A.Somayaji, S. Hofmeyr, & S. Forrest, “Principles of a Computer Immune System”, 1997, New Security Paradigms Workshop pp. 75-82

[2] C. A. Janeway Jr., “The Immune System Evolved to Discriminate Infectious Nonself Noninfectious Self”, 1992

[3] Coremann, “Introduction to Algorithms”, PHI Publications, 2nd Edition

[4] D. Stow and C. Roadknight,”Antigens, Antibodies, and the World Wide Web”, 1999

[5] D.Dasgupta, “Artificial Immune Systems and their Applications”, 1994.

[6] F.M. Burnet, “Clonal Selection and After”, 1978

[7] G.J.V. Nossal, “Negative Selection of Lymphocytes”, 1994

[8] K.Davoian, and S.Gorlatch, “A Modified Genetic Algorithm for the Traveling Salesman Problem and Its Parallelization.Proceeding Artificial Intelligence and Applications”, 2005

[9] L. N. de Castro and F.J.V. Zuben, “The Clonal Selection Algorithm with Engineering Applications”, 2002

[10] L. N. de Castro and F.J.V. Zuben, “Artificial Immune Systems: Basic Theory and Applications”, 2000

[11] L. N. de Castro, “Immune Engineering: A personal account”, 2001

[12] M.D.Mannie,”Immunological Self/Nonself Discrimination”, 1999

[13] M.Middlemiss, “Positive and Negative Selection in a Multilayer Artificial Immune System, 1997

[14] N. K. Jerne, “Idiotypic Networks and Other Preconceived Ideas”, 1984

[15] N.S. Deo, “Graph Theory and its applications”, TMH Publications, 2nd Edition

[16] S.A. Hofmeyr and S.Forrest, “Immunity by Design: An Artificial Immune System”, 1999

[17] S.Forrest, B.Javomk, R.E.Smith., and A.S.Perelson, “Using Genetic Algorithms to Explore Pattern Recognition in the Immune System. Evolutionary Computation”, 1994

[18] S. Kirkpatrick, C.D. Gelatt Jr.and M.P. Vecchi,”Optimization by Simulated Annealing”, 1987

[19] T.C. Raymond, “Heuristic Algorithm for the Traveling-salesman Problem”, 1969

Tejbanta Singh Chingtham is associated with Sikkim Manipal University since August, 2000 and is working as a Reader in the Department of Computer Science & Engineering. He received his B.Tech in Computer Science & Engineering from Bharathiar University, Coimbatore, India and M.Tech from Indian Institute of Technology Guwahati, India in Computer Science & Engineering and pursuing PhD from Birla Institute of Technology Mesra, Ranchi, India. He is an active member of IEEE, IAENG and IACSIT. He has served as Technical Committee Members of Various Journals and Conferences. He has also been listed in the Marquis Who’s Who of the World in the 2010 edition. His main areas of Interest are Artificial Immune System, Natural Computing, Evolutionary Computation and Mobile Autonomous Robots, Navigation and Path Planning.

G. Sahoo received his M.Sc. degree in Mathematics from Utkal University in the year 1980 and Ph. D. degree in the area of Computational Mathematics from Indian Institute of Technology, Kharagpur in the year 1987. He has been associated with Birla Institute of Technology, Mesra, Ranchi, India since 1988 and is currently working as Professor and Head in the Department of Information Technology. His research interest includes Theoretical Computer Science, Parallel and Distributed Computing, Evolutionary Computing, Information Security, Image Processing and Pattern Recognition.

M.K.Ghose obtained his Ph.D. from Dibrugarh University, Assam, India in 1981. He is currently working as the Professor and Head of the Department of Computer Science & Engineering at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India. Dr. Ghose also served in the Indian Space Research Organization (ISRO) – during 1981 to 1994 at Vikram Sarabhai Space Centre, ISRO, Trivandrum in the areas of Mission simulation and Quality & Reliability Analysis of ISRO Launch vehicles and Satellite systems and during 1995 to 2006 at Regional Remote Sensing Service Centre, ISRO, IIT Campus, Kharagpur(WB), India in the areas of RS & GIS techniques for the natural resources management. Dr. Ghose has conducted a number of Seminars, Workshop and Training programmes and published around 35 technical papers in various national and international journals in addition to presentation/ publication of 125 research papers in international/ national conferences. He has also provided consultancy services to many reputed institutes in India. Dr. Ghose is the Life Member of Indian Association for Productivity, Quality & Reliability, Kolkata, National Institute of Quality & Reliability, Trivandrum, Society for R & D Managers of India, Trivandrum and Indian Remote Sensing Society, IIRS, Dehradun.