Cost Reduction of Traveling Salesman Problem with an Enhanced Genetic Algorithm

K. B. Ishola[1*], O. E. James[2]

[1,2]*Department of Computer Science, Federal University of Lafia, P.M.B 146 Lafia, Nigeria*

**Corresponding Author*

Abstract: - **Traveling Salesman Problem is a variation of NP hard problem and that has made it an interesting and challenging problem in the field of computer science, even though many techniques have been proposed to improve the performance of TSP.**

Genetic Algorithm is a technique used in computing to search the optimal solution from a various possible solution to a computational problem in order that maximizes or minimizes a particular function and Travelling Salesman Problem (TSP) is computational optimization problem. The time to solve TSP grows exponentially as the number of cities increases; if it is to be solved within a reasonable amount of time then it requires optimal solution. This research work examines the solution to improve the performance of TSP by coding it into a genetic form.

The aim of this research work is to use the modified elements of Genetic Algorithm such as chromosomes, selection, crossover, mutation and fitness function to solve the Travelling Salesman Problem where one has to find the shortest or efficient route among the cities from the origin.

Keywords- **TSP, NP-Complete Problem, Genetic Algorithm, GA operator**

I. INTRODUCTION

Traveling salesman problem (TSP) is NP Complete Problem, where a salesman needs to travel, given a number of cities and the distances (costs) of traveling from any city to any other city, the goal is to find optimal or shortest route from the starting point and visits each city exactly once and then returns back to the starting city[1].

TSP as NP-Hard problem combinatorial optimization problems means the solution techniques have not been improved in polynomial time. If an efficient algorithm is found for the TSP problem, then efficient algorithms could be found for all other problems in the NP complete class. However, it has been known to required heuristic algorithms such as the genetic algorithm (GA) that could obtain possible optimal solutions within reasonable time[2].

no general method of solution is known given a large problem size

In this research project, Genetic Algorithm is used to solve the Travelling Salesman Problem, based on methods adapted from the field of genetics in biology; Genetic Algorithm is used for optimizing computer models. In Genetic Algorithms techniques, encoding of possible model behaviors is been done into "genes". For each successful generation, rating the current models allows it to mate and breed based on their fitness. In the process of mating, the genes are exchanged; so that crossovers and mutations can take place. The current population is discarded and its offspring forms the next generation subsequently.

II. RELATED WORK

The history of TSP started with Euler. TSP began from his studies "Studied the Knight's tour Problem" in 1766. According to [3], It was about discovery paths and circuits on the graph, satisfying certain conditions. Most of research into TSP history as a whole was done in the period from 1800 to 1900.

Travelling salesman problem (TSP) is a famous combinatorial optimization problem, studied in operational research and computer science. The formulated problem is as follows. Given set of n cities and distances between each pair of them, find a fastened tour through all cities that visits each city exactly once and is of minimal length. The problem is said to be NP-hard, therefore many heuristics have been proposed to find optimal solutions [4].

According to [5] Traveling Salesman Problem (TSP) is one of the broadly studied combinatorial optimization problems. TSP which is a NP hard problem is associated with determining the shortest tour through a given set of points in a space.

[6] uses genetic algorithm operators to solve the TSP problem. According to him, The G A is persistence used for improving the solution space problem. The crossover is the important stage in the genetic algorithm.

[7] solved travelling salesman problem using various genetic algorithm operators like selection methods, crossover methods and mutation methods are also mentioned in the paper. New crossover operation, population reformulation operation, multi-mutation operation, partial local optimal mutation operation, and rearrangement operations are used by [8] to solve the Traveling Salesman Problem.

Genetic algorithm operators was used to solved the traveling salesman problem to reduce the total distance and time [9]. Generating the fittest criteria using selection, crossover and mutation operators has made it possible with a new crossover method; the Sequential Constructive Crossover method is used.

III. METHODOLOGY

Genetic Algorithm as an intelligence method is a search technique used to find approximate solutions to optimization problems. GA is more suitably said to be an optimization technique that embrace the survival of the fittest idea algorithm [10]. The idea is to first the solutions and then combining the fittest solution to produce a new set of solutions better than the previous generation.

A simple GA works by generating an initial population of strings randomly, which is referred as gene pool and then applying (possibly three) operators to create new, and hopefully, better populations as successive generations. The first operator is reproduction where strings are copied to the next generation with some probability based on their objective function value. The second operator is crossover where randomly selected pairs of strings are mated, creating new strings. The third operator, mutation, is the occasional random alteration of the value at a string position. The crossover operator together with reproduction is the most powerful process in the GA search.

Parameters of Genetic Algorithm

Chromosome

A set of genes; chromosome comprises the solution in the form of genes. Gene is a part of chromosome that encloses a part of solution or determines the solution. For example, 1000011 can be a chromosome and its equivalent genes are 1,0,0,0,01,1.

Chromosome coding is the first stage of problem solving using genetic algorithm. This coding may vary from one problem to the other. Binary coding and Permutation coding are the most common ones. Each chromosomes is represented by a sequence of characters (0and 1) while in permutation coding each chromosomes is represented byarranging a series which is the most common in Traveling Salesman Problem.

Population Size

Number of chromosomes in population. This largely depends on the type of encoding and the problem. If the chromosomes are few then GA would have a few likelihoods to implement crossover and only a small portion of search space will be explored. If the chromosomes are much then GA would slows down.

Reproduction

Usually, it is the first operator applied on population. Good Individuals must have continuity life, there is need to create new individual from the group to create a new population by crossover and mutation. By Darwin‘s evolution, chromosomes are selected from the population of parents to cross over and produce offspring. This operator is also known as Selection Operator‘.

Cross Over

After the completion of reproduction phase, population is enhanced with an improved individual. It makes clones of good strings but does not create new ones. Cross over operator is applied with a hope that it would create better strings and best chromosomes for the next cross over step.

Crossover Methods

Single Point Crossover: Single point crossover is a universally used method of crossover. In this method, a position is chosen at random and the fragments of two parents after the crossover position are swapped to produce two offspring.

Parent 1: 1 1 0 | 0 0 1

Parent 2: 0 1 0 | 1 1 1

Offspring 1: 1 1 0 1 1 1

Offspring 2: 0 1 0 0 0 1

Multipoint Crossover: It is a larger view of single point crossover, presenting a higher number of cut points.

Parent 1: 1 0 0 |1 0 1| 1 1

Parent 2: 0 1 1 |1 0 0 | 0 1

Offspring 1: 1 0 0 1 0 0 1 1

Offspring 2: 0 1 1 1 0 1 0 1

Uniform Crossover: It uses a global parameter to indicate the probability that each variable should be swapped between two parents.

Parent 1: 1 0 1 0 0 0 1 1 1 0

Parent 2: 0 0 1 1 0 1 0 0 1 0

Offspring 1: 0 0 1 1 0 0 1 0 1 0

Offspring 2: 1 0 1 0 0 1 0 1 1 0

Mutation

Immediately after cross over, the strings are exposed to mutation. Mutation operator permits new individual to be created by selecting them based on their fitness value. Chromosomes involved are randomly chosen to mutate and switch that point 0 to 1 and vice-versa for a binary chromosome representation using bit flip method.

Fitness Function

Genetic algorithms are used for maximization problem. For the maximization problem the fitness function is same as the objective function. But, for minimization problem, one way of defining a ‘fitness function’ is to evaluate the fitness f(x) of each chromosome x in the population

$$F(x) = \frac{1}{f(x)} (1)$$

Where f(x)is the objective function that calculates cost (or value) of the tour represented by a chromosome Since TSP is minimization problem.

Pseudo Code of Genetic Algorithm

Begin GA

INITIALIZE population with random candidate suitable solutions for the problem;

EVALUATE the fitness *f(x)* of each candidate in the population;

Repeat

SELECT two parent from a population according to the fitness;

RECOMBINE pairs of parents to produce a new offspring (children);

MUTATE the resulting children at each locus (position in chromosome);

EVALUATE children (new offspring) in a new population;

SELECT individuals for the next population for a further run of algorithm;

Until

TERMINATION-CONDITION is satisfied, and then stops the algorithm and output the best solution in current population;

End

Use Cases of TSP with GA

The use case diagram illustrates the activities that are performed by the user of the system. Below are use cases for the system

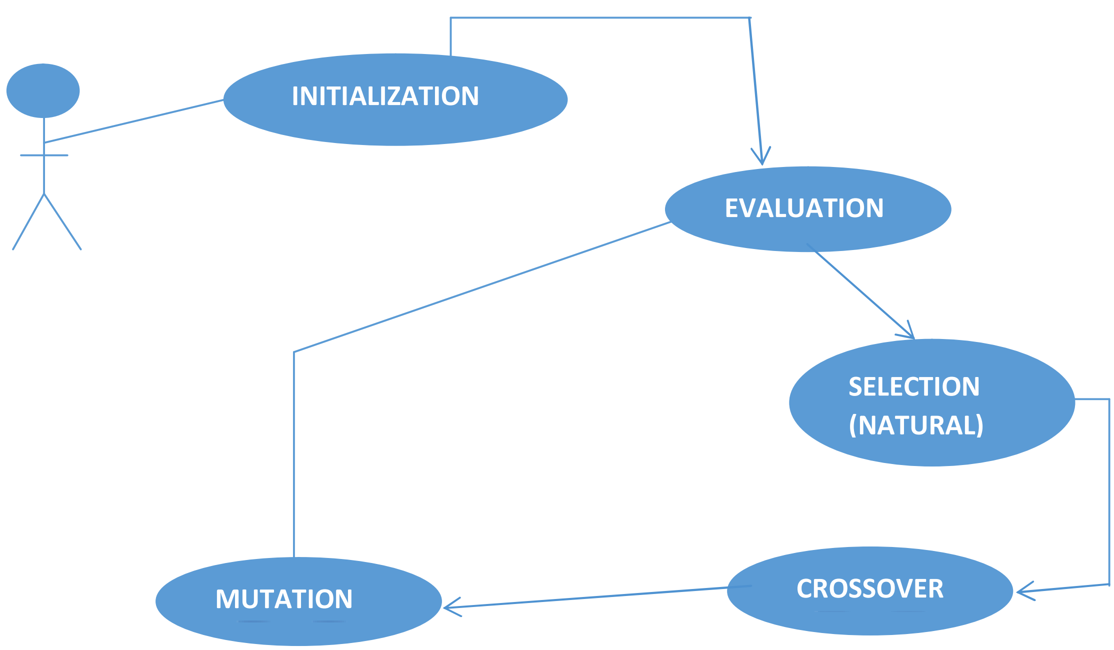

Fig 3.1: Genetic Algorithm use case

Flow Diagram

This is used to model system components. These components include input data to the system, various processing carried, external entity that interacts with the system and the information flow in the system

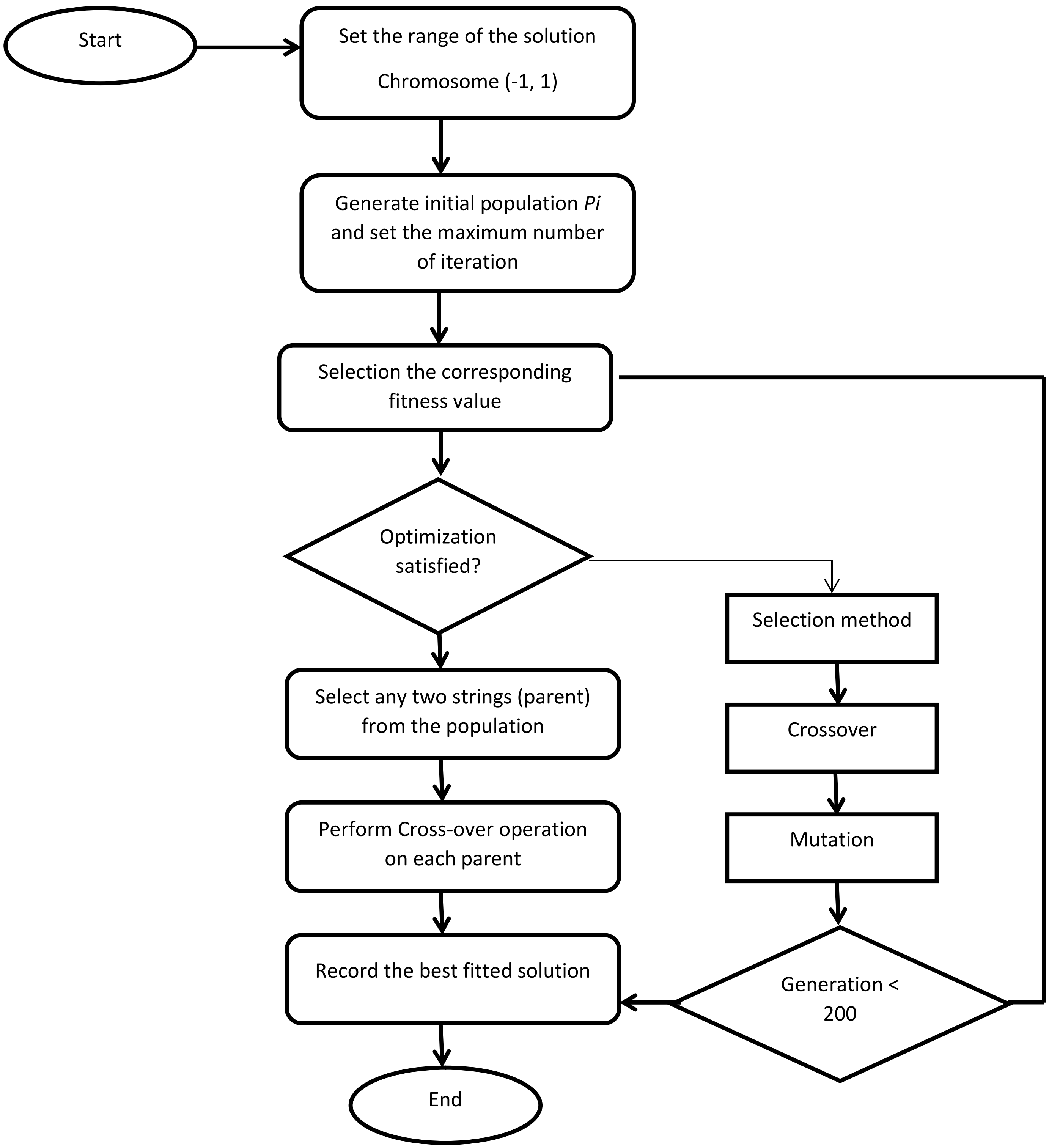

Figure 3.2: Genetic Algorithm process flow diagram

IV. RESULTS AND DISCUSSION

Definition

Traveling Salesman Problem is an exceptionally indispensible problem in operational research, there is need for us to define and understand the problem then implement an algorithm to solve TSP [11].

There is a group of distinct cities$\{V_1,\ V_2, \ldots V_n\}$and there is a corresponding edge for each pair of cities$\{V_i,\ V_j\}$and a closedpath $V_\alpha\ =\ \{V_{\alpha(1)},\ V_{\alpha(2)}, \ldots \{V_{\alpha(n)}\}$.The objective is to find an ordering of cities such that the total time for the salesman is minimized. The lowest possible time is called the optimal time. The objective function is given as:

$$\sum_{i=1}^{N-1} d\left(V_{\alpha(i)}, V_{\alpha(i+1)}\right) + \ d\left(V_{\alpha(N)}, V_{\alpha(1)}\right) \ (2)$$

Traveling Salesman Problem can be categorized in two classes: Symmetric and Asymmetric TSP

Symmetric Travelling Salesman Problem

The distance between two edges is always equal in each opposite direction, forming an undirected graph. Let V = $\{(v_1, \ldots v_2)\}$ be a set cities, A = {(r, s): r, s ε V} be the edge set, and d_{rs}= d_{sr}be a cost measure associated with edge (r, s) ε A.

Asymmetric Travelling Salesman Problem

The distances between two edges might be different, forming a directed graph and path may not exist in both directions. If $d_{rs} \neq d_{sr}$for at least one (r, s), then the TSP becomes Asymmetric TSP

Mathematical Formulation of TSP

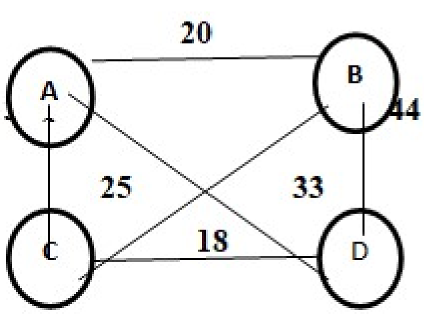

Figure 4.1: TSP as a graph problem

TSP as a graph problem

Let a graph G = (V, A) where V is a set of n vertices, A is a set of edges, and let C: C_{ij}be a distance matrix associated with A. The TSP lies on determining a least distance circuit passing through each vertex once and only once.

TSP can be modeled as an undirected weighted graph, such that cities are the graph's vertices (V), paths are the graph's Edges (E), and a path's distance is the edge's length. Often, the above graph is a complete graph (i.e. each pair of vertices is connected by an edge).

Mathematical Modeling

Mathematically, the objective of this problem is to find a tour, among all the possible tours, that reduces the total distance the salesman needs to travel. Whereas each city should be visited just only once and returns back to the start city.

Tag the cities with the numbers *0,…, n*. For n cities to visit, let x_{ij} be the variable that has a value 1 if the salesman goes from city i to city j and a value 0 if the salesman does not go from city i to city j. Letd_{ij}be the distance from city i to city j.

Then the TSP can be stated as

Minimize the linear objective function:

$$Z = \sum_{i=0}^{n} \sum_{j\neq i\ j=0}^{n} x_{ij} d_{ij} \qquad (3)$$

$0 \leq x_{ij} \leq 1 \qquad i, j = 0, \ldots, n$

Subject to the constraint,

$$\sum_{i=0,\ \ i\neq j}^{n} x_{ij} = 1 \qquad j = 0, \ldots n \qquad (4)$$

$$\sum_{i=0,\ \ j\neq i}^{n} x_{ij} = 1 \qquad i = 0, \ldots n \qquad (5)$$

Development Environment

While there are a variety of environments in which this type of Research work could be done, but this work was implemented on a windows operating system, with the use of Netbeans IDE.

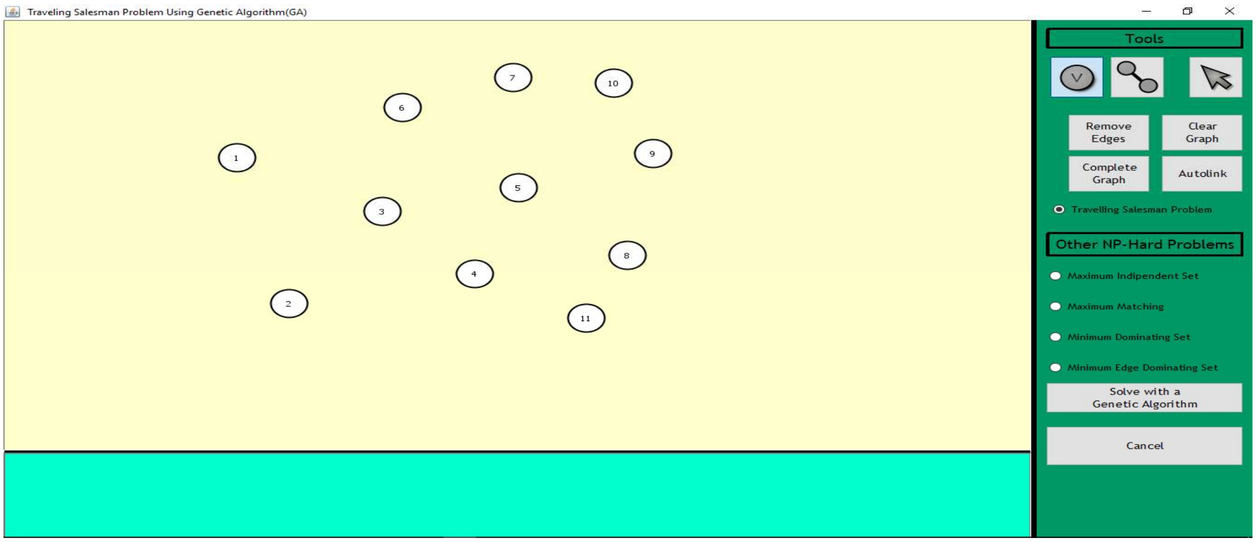

Figure 4.2: Generated Cities

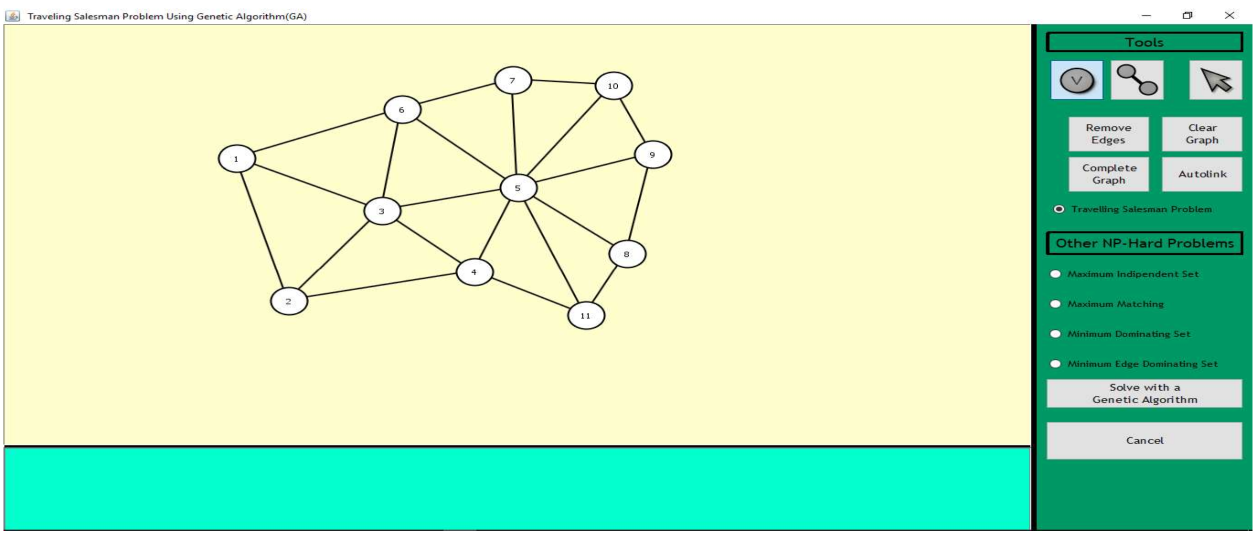

Figure 4.3: Connected Cities

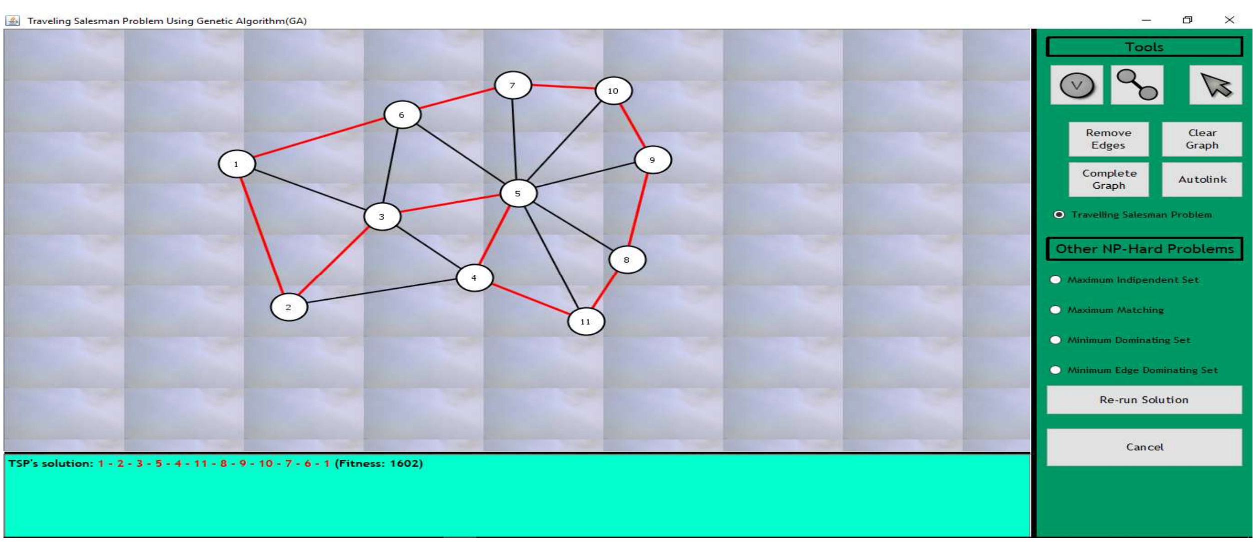

Figure 4.4: Traveling Salesman Solution

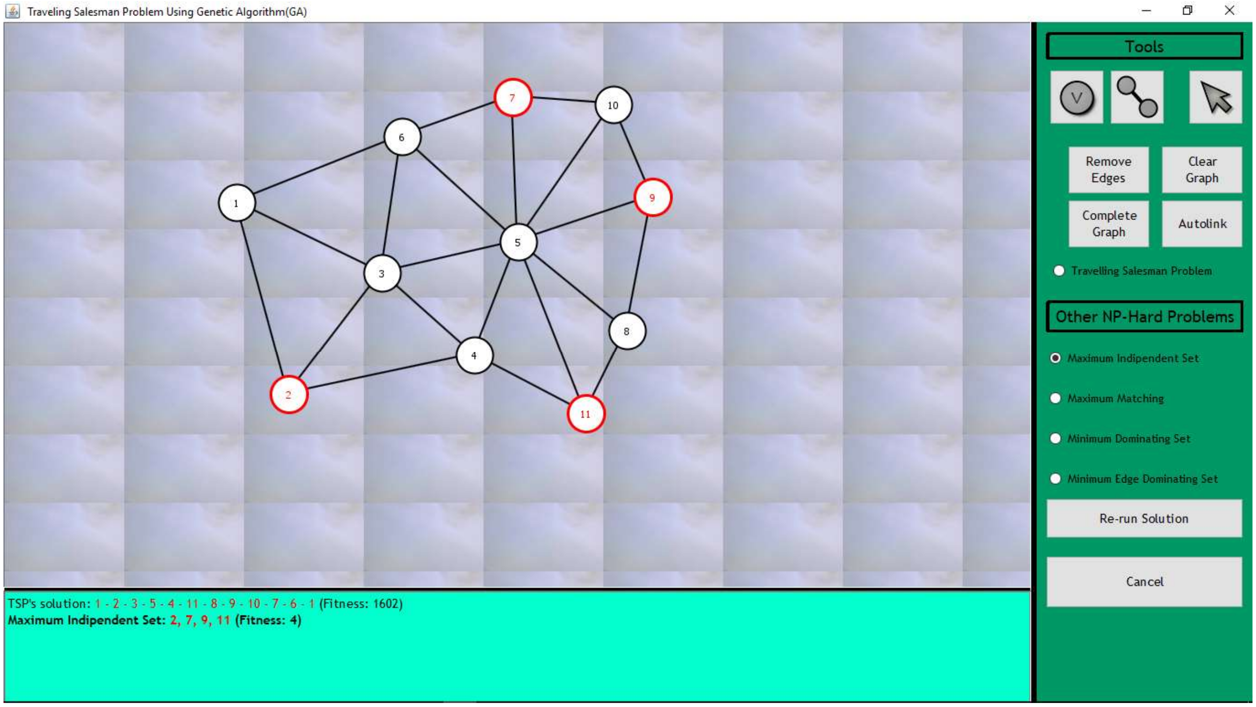

Figure 4.5: Maximum Independent Set Solution

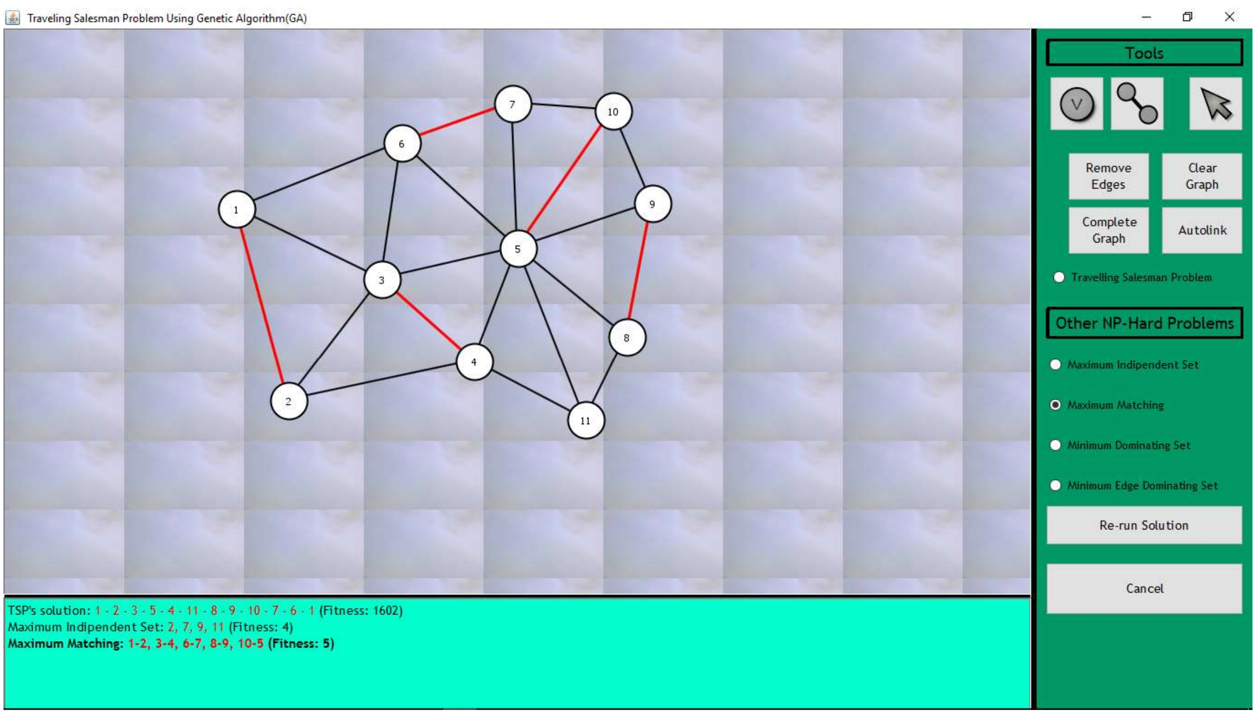

Figure 4.6: Maximum Matching Solution

V. CONCLUSION AND FUTURE SCOPE

In conclusion, this system that was established using JAVA and buildup Genetic Algorithm library provides a platform for the users pinpoint different nodes (cities) as much as possible and link them with one another as so desire and shows that the shortest possible path or routeto the final destination from the starting point. The system is efficient, faster and reliable. It can also perform the following:

- Find the minimum dominating sets between the connected cities
- Maximum independent set in the generated cities so on.

To make improvements to the system, some work will be necessary in order to enhance and make the application more flexible for users to specify from the interface, the number of time (iteration) it uses to compute the connected nodes, specify the number of cities/nodes to be generated itself and the total distance between each node/cities. Secondly, the system can also be extended for more sophisticated optimization algorithm. This will make the application more usable and versatile.

REFERENCE

[1] Zelinka, R., Singh, S. P. and Mittal, M. L. (2010). "*Traveling Salesman Problem*": An Overview of Applications, Formulations, and Solution Approaches, readings on the Traveling Salesman Problem, Theory and Applications, Edited by Donald Davendra.

[2] Zambito, L., D. (2006). "*The Travelling Salesman Problem*": A Comprehensive Survey. Collecting Primary Data Using Semi-Structured and In-Depth Interviews.

[3] Laporte, G. (2006). "*A Short History of the Traveling Salesman Problem*". Canada Research Chair in Distribution Management, Centre for Research on Transportation and GERAD HEC Montreal, Canada.

[4] Lawler, E. L., Lenstra, J. K., RinnooyKan, A. H. G., Shmoys, D. B. (1985). "*The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*" John Wiley & Sons.

[5] Aybars Uğur, Serdar Korukoğlu, Ali, Calıskan, Muhammed Cinsdikici, (2009) "*Genetic algorithm based solution for TSP on a sphere*", Mathematical and Computational Applications,Vol. 14, No. 3, pp. 219-228.

[6] Varshika Dwivedi, Taruna Chauhan, Sanu Saxena and Princie Agrawal (2012)."*Travelling Salesman Problem using Genetic Algorithm*", National Conference on Development of Reliable Information Systems, Techniques and Related Issues (DRISTI).

[7] Naveen kumar, Karambir and Rajiv Kumar (2012)."*A Genetic Algorithm Approach to Study Travelling Salesman Problem*" .Journal of Global Research in Computer Science, Volume 3, No. 3, ISSN-2229-371X.

[8] Omar M. Sallabi and Younis EI-Haddad (2009)."*An Improved Genetic Algorithm to Solve the Traveling Salesman Problem"*, World Academy of Science, Engineering and Technology Volume 3.

[9] Arananayakgi, A. (2014) "*Reduce Total Distance and Time Using Genetic Algorithm in Traveling Salesman Problem*", International Journal of Computer Science & Engineering Technology, ISSN : 2229-3345, Vol. 5 No. 08.

[10] Mouhammd Al kasassbeh, Ahmad Alabadleh and TahsenAl-Ramadeen, (2012). "*Shared Crossover Method for Solving Traveling Salesman Problem*", IJICS Volume 1, Issue.

[11] De Jong. (1993). "*Heuristic Combinatorial Optimization for the Traveling Salesman Problem*". Biological Cybernetics 65: 31–35.