ATM PROBLEM SOLVED BY TSP

Edra Fresku[1]
Arinela Anamali[2]

[1]IT Department, Regional Hospital of Shkodra, Albania, e-mail: freskuedra@hotmail.com
[2]IT Department, AgnaGroup, Tirana, Albania, e-mail: arinela_anamali@yahoo.com

Abstract

The traveling salesman problem (TSP) is one of the most common problems in combinatorial optimization. A number of prominent researchers have tried to attack this problem. The role of the TSP in the field is underlined by the fact that it is commonly accepted as the representative combinatorial optimization problem. Its practical importance is one of the reasons of such status. Our purpose in this term project is to implement heuristic algorithms and compare and evaluate their respective computational efficiency. Included in this model are 2-opt and random improvements algorithms.

The problem that we want to analyze is: A bank has many Automated Teller Machine (ATM) machines. Each day, a courier goes from machine to machine to make collections, gather computer information, and service the machines. In what order should the machines be visited so that the courier's route is the shortest possible? This problem arises in practice at many banks. One of the earliest banks to use the TSP algorithm, in the early days of ATMs, was the Shawmutt Bank in Boston.

The objective of this study is to solve the ATM problem using travelling salesman problem (TSP) with different algorithm that are 2-opt and Random Improvements algorithm. For analyzing the result we have taken in consideration three criteria: number of ATM machines, best result, and the time to solve this problem. The objectives are:

- To use the TSP to solve ATM problem for various approaches such as:
  - 2-opt algorithm
  - Random Improvements algorithm
- To analyze 2-opt and Random Improvements algorithms for solving ATM problem
- To compare and evaluate their respective efficiency

Result of the comparison is that the performance and efficiency of 2-opt algorithm is better than Random Improvement.

Keywords:*automated teller machine (ATM), travelling salesman problem(TSP)*

1. INTRODUCTION

The traveling salesman problem (TSP) is one of the most common problems in combinatorial optimization. A number of prominent researchers have tried to attack this problem. The role of the TSP in the field is underlined by the fact that it is commonly accepted as the representative combinatorial optimization problem. Its practical importance is one of the reasons of such status. In fact, many significant real world problems can be formulated as instances of the TSP. The well known applications of the TSP include vehicle routing, circuit wiring, network connection, job sequencing. Also, the areas where TSP has had a profound influence are: Operational Research, Discrete Mathematics, Theoretical Computer Science, and Artificial Intelligence.

The definition of the TSP can be simply stated without any mathematical notation as follows. A salesman has to visit n cities once and only once and finish where he started. Given the cost of travel between each pair of cities, the salesman wants to find the minimum cost tour of cities. At a glance, the salesman's problem looks very straightforward and easy. However, the difficulty is revealed if the number of possible tours is considered.

In 1979, Garey and Johnson proved that TSP is an NP-hard problem that cannot yet be solved in polynomial time. Though without a proof that $P \neq NP$, it can only be assumed that there is no efficient algorithm for solving any version of the TSP. Thus it is infeasible to follow complete enumeration of large size real-world TSP instances. Even if there is an exact method that guarantees an optimal solution, its running time is prohibitively excessive for large-scale problems. In order to handle such large problems, many heuristic methods have been developed. These heuristic methods can quickly find solutions of acceptable quality.

- **Versions of TSP**

The TSP has been most commonly expressed in two forms:

1. The combinatorial optimization version i.e. the problem of finding the minimum Hamiltonian cycle in a graph of cities
2. The decision version i.e. the problem of checking the existence of a Hamiltonian cycle in a graph smaller than a given weight.

- **Classification of TSP**

Different instances of the TSP are also divided into different classes based on the arrangement of distance between the cities or the type of graph in concern. In the Symmetric TSP, the distance between two cities is the same in each direction, forming an undirected graph. This symmetry halves the number of possible solutions. In the Asymmetric TSP, paths may not exist in both directions or the distances might be different, forming a directed graph.

2. SOLVING THE TRAVELLING SALESMAN PROBLEM

There have been many efforts to solve the travelling salesman problem ever since it was coined in 1930. The deterministic approaches strive to give an exact solution to the problem whereas the non deterministic try to provide a tour with a near minimal cost.

- **Deterministic Approaches to TSP**

One of the earliest deterministic solutions to TSP was provided by Dantzig et al., in which linear programming (LP) relaxation is used to solve the integer formulation by adding suitably chosen linear inequality to the list of constraints continuously. Held and Karp presented a dynamic programming formulation for an exact solution of the travelling salesman problem; however it has a very high space complexity, which makes it very inefficient for higher values of N.

The branch and bound technique based algorithm published in was able to successfully increase the size of the problem solvable without using any problem specific methods. The algorithm branches into the set of all possible tours while calculating the lower bound on the length of the tour for each subset. Eventually it finds a single tour in a subset whose length is less than or equal to some lower bound for every tour. The algorithm however grows exponentially in time with the input, but it is able to calculate the TSP for 40 cities with appreciable average time consumption as displayed by the authors.

In the survey on exact algorithms for the Travelling Salesman problem, most attempts were found trying to address just a subset of the problem, instead of working on the complete problem space. This approach proved to be successful in almost all such cases, often slightly advantageous as far as time complexity is concerned. This happens due to the fact that not all problem instances are equally difficult. An efficient solution to the specific class of such instances can be used extensively by applications which deal mostly with these relatively easy instances only.

Other exact solutions to the problem as described in include **cutting plane method** (introduced by Dantzig, Fulkerson, and Johnson), branch-and-cut (introduced by Padberg and Rinaldi), branch-and-bound (Land and Doig) and Concorde (introduced by Applegate, Bixby, Chatal, and Cook). Although each of these algorithms become highly inefficient as we increase the number of cities in the tour.

- **Non-Deterministic Solution to TSP**

The exact solutions provide an optimal tour for TSP for every instance of the problem; however their inefficiency makes it unfeasible to use those solutions in practical applications.

Therefore Non-Deterministic solution approach is more useful for the applications which prefer time of run of the algorithm over the accuracy of the result. There has been a vast research in past to solve the TSP for an approximate result. Some of the implemented approximate algorithms are being listed here.

- **Nearest Neighbor Algorithm**

It follows a very simple greedy procedure: The algorithm starts with a tour containing a randomly chosen city and then always adds to the last city in the tour the nearest not yet visited city. The algorithm stops when all cities are on the tour.

- **Insertion Algorithms**

All insertion algorithms start with a tour consisting of an arbitrary city and then choose in each step a city k not yet on the tour. This city is inserted into the existing tour between two consecutive cities i and j, such that the insertion cost (i.e., the increase in the tour's length) $d(i; k) + d(k; j) = d(i; j)$ is minimized. The algorithms stop when all cities are on the tour.

- **K-Opt Heuristics**

The idea is to define a neighborhood structure on the set of all admissible tours. Typically, a tour t is a neighbor of another tour t if t can be obtained from t by deleting k edges and replacing them by a set of different feasible edges (a k-Opt move). In such a structure, the tour can iteratively be improved by always moving from one tour to its best neighbor till no further improvement is possible. The resulting tour represents a local optimum which is called k-optimal.

Researchers have also taken help from literature on artificial intelligence and machine learning. Various algorithms optimizing the power of neural networks have been proposed for the approximation of the optimal cycle. Also a popular method is the Ant Colony optimization scheme.

3. FASTEST KNOWN SOLUTIONS TO TSP

- **Deterministic**

The Concorde method proposed by Applegate D, Bixby RE, Chvatal V, Cook W is widely regarded as the fastest TSP solver, for large instances, currently in existence. In 2001, it even won a 5000 Guilder prize from CMG for solving a vehicle routing problem the company had posed in 1996. It holds a record of solving a TSP instance having 15,112 nodes. However it has the capability of solving only symmetric versions of the problem.

- **Non-Deterministic**

The Lin-Kernighan heuristic is generally considered to be one of the most effective methods for generating optimal or near-optimal solutions for the symmetric traveling salesman problem (TSP). The heuristic grows by a relation of n2 with time as shown in his paper by W. Kernighan. However, the design and implementation of an algorithm based on this heuristic is not trivial. There are many design and implementation decisions to be made, and most decisions have a great influence on performance.

4. SOLVING ATM PROBLEM USING TSP

Various heuristics and approximation algorithms, which quickly yield good solutions, have been devised. Modern methods can find solutions for extremely large problems (millions of ATM machines) within a reasonable time which are with a high probability just 2-3% away from the optimal solution.

Several categories of heuristics are recognized.

- **Constructive heuristics**
- **Iterative improvement**
  - Pair wise exchange, or Lin–Kernighan heuristics
  - k-opt heuristic
  - V-opt heuristic

- **Randomized improvement**

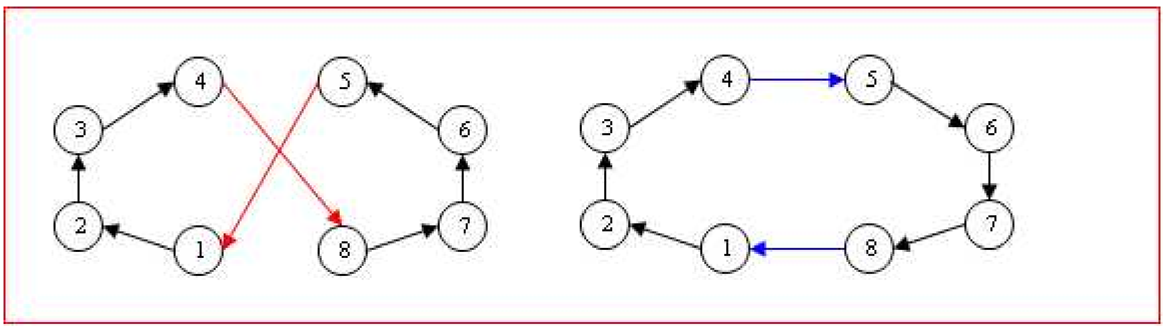
To solve the ATM problem we have chosen Heuristic Algorithms:

1. 2-Opt algorithm
2. Random Improvement Algorithm

The reason that we have chosen these algorithms, it's because these algorithm can be improved, in order to improve their results.

- 2-OPT Algorithm

2-Opt is probably the most basic and widely used local search heuristic for the TSP. It is included in Non - Deterministic Solution. This heuristic achieves amazingly good results on "real world" with respect to running time and approximation ratio. Observe that the 2-opt algorithm considers only pair wise exchange. Randomly pick two links between cities in our best random solution. Then remove those links and replace them with others that keep the route connected.



In this illustration, we remove the red links 5-1 and 4-8 on the left, and replace them with the blue links 4-5 and 8-1 on the right. Notice that we need to reverse some of the other links to keep the route connected.

Next we check if this change improves the overall distance traveled. If it does, we use the new ordering and try swapping other random links. We keep going until we have tried a large number of swaps with no improvement, at which point we stop. The solution we end with may not be optimal, but it's a very good one.

This heuristic can be generalized to k-opt, that is, you can rearrange three, four, or more links in the same way we did two. The more links you remove, however, the more complicated it is to put the route back together again so it's connected.

- Randomized improvement Algorithm

A very straightforward approach to solving TSP is to simply pick a bunch of random orderings for the stops and then see which one is best.

Algorithm of Random Approach:

1. Visit (start at) the start ATM machine.
2. As long as some ATM machines has not been visited:
   a) Pick one of the unvisited ATM machines at random.
   b) Visit that ATM machine.
3. Return to the start ATM machine.

Instead of just making a bunch of random solutions, we improve this algorithm significantly if we try to improve the random solution. Rather than testing a random guess and moving on to the next one, we try making small changes to the random guess to see if such changes improve it.

Randomized improvement

Also this algorithm is Non - Deterministic Solution there are several ways we can try to improve a solution. One of the easiest is to randomly pick two stops, swap them, and see if that made the route better. If the route is better, we keep the change and try to make other improvements. If the swap didn't improve the solution, we move the stops back to their original positions and try again with two other stops.

Random path change algorithms are currently the state-of-the-art search algorithms and work up to 100,000 cities. The concept is quite simple: Choose a random path, choose four nearby points, and swap their ways to create a new random path, while in parallel decreasing the upper bound of the path length. If repeated until a certain number of trials of random path changes fail due to the upper bound, one has found a local minimum with high probability, and further it is a global minimum with high probability (where high means that the rest probability decreases exponentially in the size of the problem - thus for 10,000 or more nodes, the chances of failure is negligible).

- **Design & Implementation**

To design and to implement this project we have used c# programming languages. The environment where we have developed this project is Visual Studio .NET 2005.

The main menu of application is as follow:

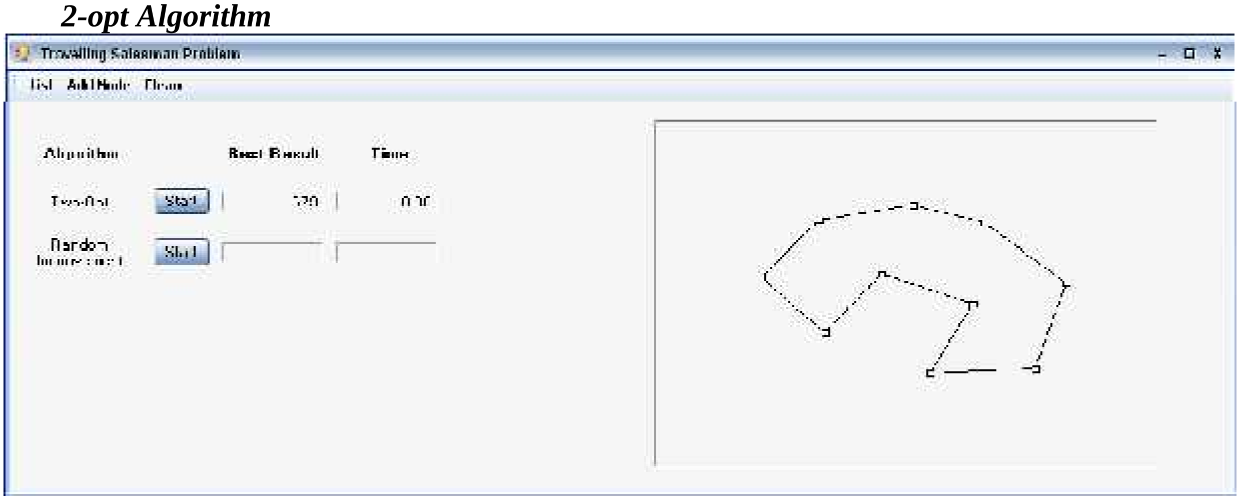
Add Node menu can add by clicking on the panel all the point of ATM machines that we want.

Clean menu clean the panel.

Start button run the algorithm.

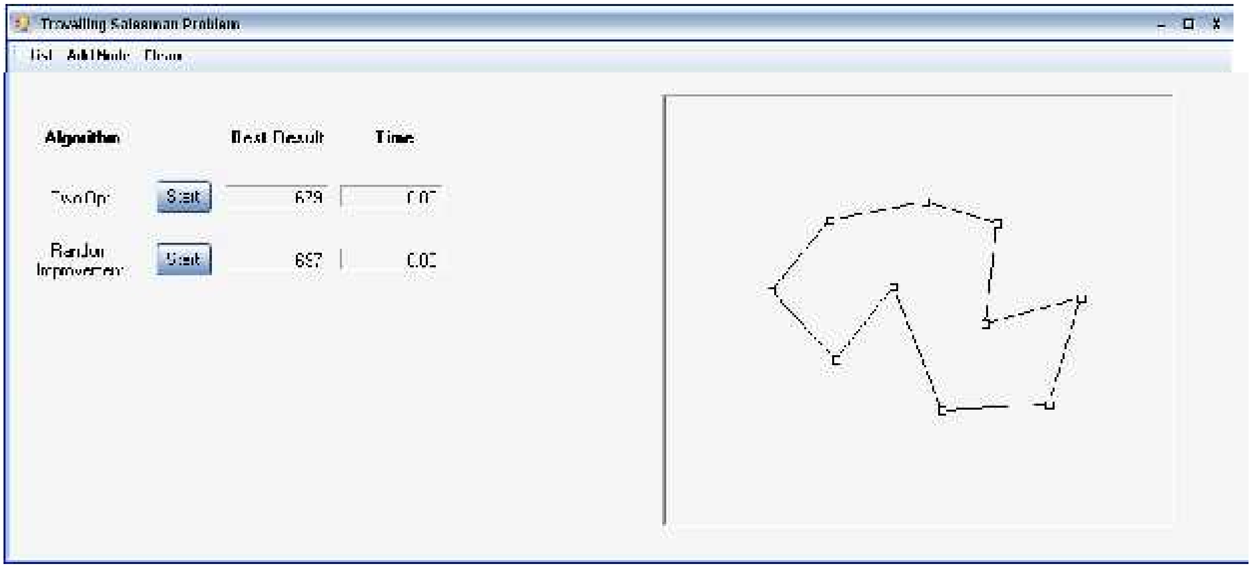
We have implemented both algorithm, and display the best length of the route that can do the courier for visit all ATM machines. Also display the time that is needed to perform this route.

2-opt Algorithm



Length: 679
Time: 0.00

Random Improvement Algorithm



Length: 697
Time: 0.00

We did a performance comparison of 2 algorithms.

As we mentioned before, we have taken in consideration three criteria to compare both algorithm:

- number of ATM machines
- best result
- time to find the shortest route

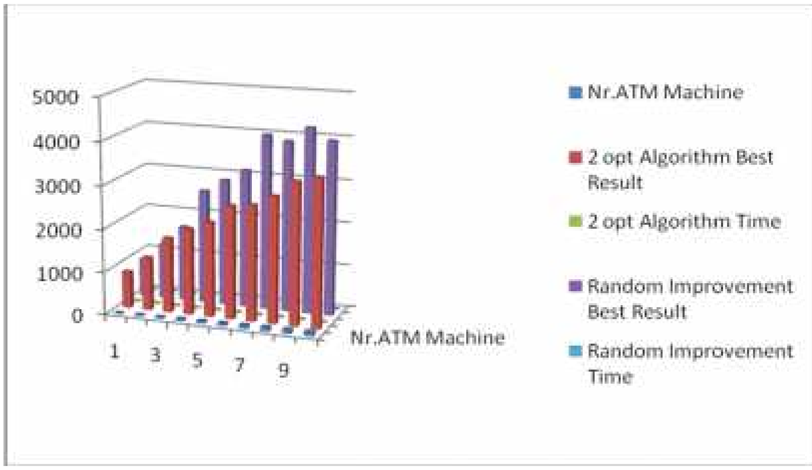
For the simulation ATM machines size of 10 to100 are used. The below tables and graphs illustrate the result of simulation.

The table below compare the result for the 2-opt Algorithm and Random Improvement Algorithm:

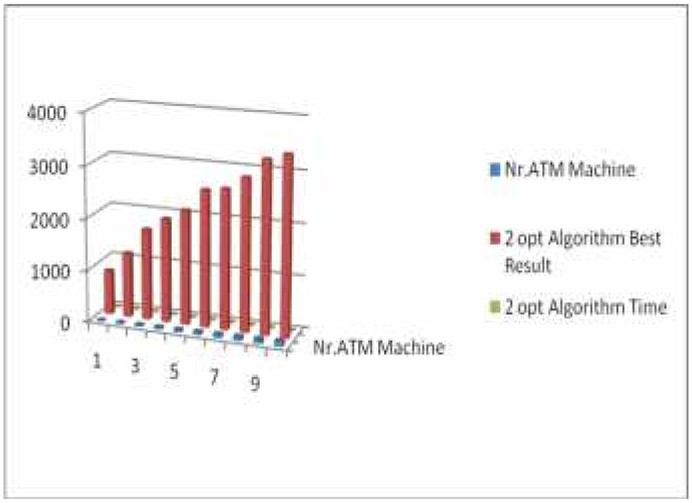
Nr.ATM Machines	2 opt Algorithm		Random Improvement	
	Best Result	Time	Best Result	Time
10	857	0	890	0
20	1247	0	1301	0
30	1754	0	1749	0
40	2000	0	2654	0
50	2205	0	2945	0
60	2630	0	3216	0.02
70	2690	0	4060	0.02
80	2936	0.02	3951	0.03
90	3305	0	4289	0.02
100	3435	0	4027	0.06

- **Algorithms Comparison**

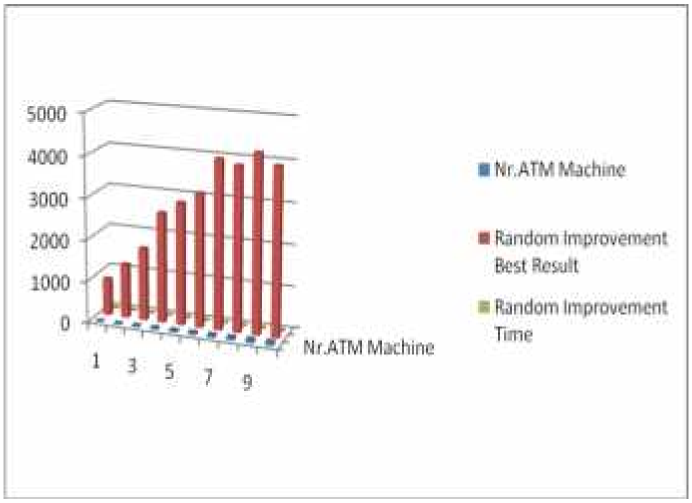
Length and time comparison on 2-opt Algorithm and Random Improvement Algorithm.



Length Comparison on 2-opt Algorithm



Time Comparison on Random Improvement



From the chart we can see that: When the number of ATM machines increase, also increase the length of the route. For the last 4 group of ATM machines the Random Improvement Algorithm find a route higher then 2-Opt Algorithm. So, when the number of ATM machines increase, for 2-opt algorithm the length of the route increase, also for Random Improvement Algorithm the length of the route increase, but this increase is much higher on Random Improvement Algorithm.

5. CONCLUSION & FUTURE

The Traveling Salesman Problem (TSP) is perhaps the most studied discrete optimization problem. Its popularity is due to the facts that TSP is easy to formulate, difficult to solve, and has a large number of applications.

According to the result of algorithms, as we can see from the chart, if the number of the ATM machines is increased the simulation time also increased for Random Improved algorithm.

For 2-OPT Algorithm the simulation time in general is the same, although we increase the number of ATM machine.

For small-size TSP (n < 50), both algorithm have the same simulation time, but the length of route is smaller in 2-opt algorithm then in Random Improved algorithm. For this range 2-opt algorithm is recommended.

For large size problem (n > 50), Random Improved algorithm has the simulation time more higher than 2-opt algorithm. Also for this range, the improved genetic algorithm is recommended.

So the performance and efficiency of 2-opt algorithm is better than Random Improvement.

Few researches have been done on other variants of the TSPs, which shows a potential future direction for research. More complicated encoding scheme, crossover and mutation operators are required for other variants of the TSPs, which is surely of great help to extend the applications of Genetic Algorithm. The comparison of GA, Simulated Annealing, heuristics, and other global optimization arts is also helpful for us to understand the capacities of these methods.

REFERENCES

Byung-In Kim, Jae-Ik Shim, Min Zhang, *"Comparison of TSP Algorithms"*, December 1998

Radislav Vaisman, *"TSP random tour generation algorithm"*

L´aszl´o H´ars, "Random Search in the Traveling Salesman Problem", December 1989

Ch. Engels, B. Manthey, *"Average-Case Approximation Ratio of the 2-Opt Algorithm for the TSP"*, Saarland University, Department of Computer Science

Ch. Nilsson, *"Heuristics for the Traveling Salesman Problem"*, Link¨oping University

J. Brest, J. Zerovnik*y*, "A Heuristic for the Asymmetric Traveling Salesman Problem", 2000

D. S. Johnson, L. A. McGeoch, "The Traveling Salesman Problem: A Case Study in Local Optimization", 1995

D. Applegate, R. Bixby, V.Chvatal, W. Cook, *"On the Solution of Travelling Salesman Problem"*, 1991

http://iris.gmu.edu/~khoffman/papers/trav_salesman.html

G. Gutin, "The Traveling Salesman Problem", Department of Computer Science