

An Application of Metric Method of Solving Travelling Salesman Routing Problem to Table Water Company

Akpan, N. P¹, Onyebuchi, U.R².

^{1,2}Department of Mathematics and Statistics, University of Port Harcourt, Nigeria.

nsipaulakpan@gmail.com¹, onyebu57@gmail.com²

Abstract

Background: This research work deals with the travelling salesman problem, the aim of this research is to obtain the cheapest possible route to be taken during the distribution of product (water) in some districts in Obio-Apkor Port Harcourt by Junac Table Water Company in order to maximize the company's profit.

Materials and Methods: The metric method is adopted in solving the problem analytically, and the result was also confirmed and compared with aid of a statistical software (TSP solver and Generator v0.1).

Results: The route with the minimum cost from both the matrix method and the statistical software is N1010 (one thousand and ten naira only).

Conclusions: The cheapest route to be taken is

Choba → Mbuogba → Location → Mile III → Mile I → Rumuola → Artillery → Rumuokurusi →

Tank → Eneka → Eliozu → Rumuodara → Rumuokwuta → Rumuigbo → Rumuokoro → Rumuosi → Choba.

Keywords: Transportation, Travelling salesman problem, routing, genetic algorithm, greedy algorithm, metric method.

I. Introduction

The problem consisting a salesman and a set of cities in which the salesman has to visit each one of the cities starting from a certain city and returning back to the same city he started is known as the Traveling Salesman Problem (TSP). Knox [5] notes that an intriguing aspect of the TSP is the relative ease with which it can be described and the extreme difficulty it presents in finding the optimal solution. In a TSP, the connections between the cities are called edges. Each edge has an assigned cost or distance which represents the cost and distance of traveling between two cities connected by an edge. The objective of this problem is to find the cheapest route, shortest distance or quickest time to travel to all the cities only once and return back to where he started. The TSP can be classified into symmetric and asymmetric TSP. The symmetric TSP deals with the situation whereby the cost (distance) of an edge is independent of the direction of travel (i.e., the cost (distance) of traveling from city A to city B is always the same as traveling from city B to city A for any pair of cities A,B). In the asymmetric TSP the cost (distance) of an edge may be dependent on the direction of travel along the edge (i.e., the cost (distance) of traveling from city A to city B may be more than the cost of traveling from city B to city A)[5]. Other types of the Traveling Salesman Problem includes the Multiple Traveling Salesman Problem (mTSP) and the Modified Traveling Salesman Problem (MTSP). The Traveling Salesman Problem has been widely studied as it is a great tool in the global market trade especially through logistics and supply chain management[10][4]. It can be applied in different ways to solve different problems as shown in the literature. Different methods have been used to solve the Traveling Salesman Problem for instance, Fereidouni[3] developed a linear programming algorithm for solving the TSP; Moon et al[8], Alhamdy et al.[2], Sadiq[12], MohdRazali[7] used the generic algorithm to solve TSP problems while Stencek[13], Ochiai&Kano[9] combined different methods to find an optimal solution. In this present study, a TSP on vehicle routing problem where a salesman has to drive through different routes to deliver a product to his customers across different cities with the objective of finding the best route for the truck so that cost of distribution is minimized, is considered. A metric method was adopted to solve this problem. The study examines a table water company (JUNAC Table Water, Choba Port Harcourt) in Nigeria whose aim is to minimize cost of production, packaging and distribution in order to maximize profit. The rest of the paper is organised as follows: Section two deals with some literatures on the Traveling Salesman Problem; Section 3 explains the methodology used in analysing the data; Section 4 presents the data; Section 5 shows the analysis using the metric method and the results and finally, Section 6 concludes the study.

II. Literature Review

Lin & Kernighan [[6]] developed a highly effective heuristic algorithm for the symmetric traveling salesman problem. They tested their methods on some 'classical' problems and randomly generated test

problems, up to 110 cities obtaining optimum solutions. Knox[[5]] described the tabu search heuristic method and applied it to the symmetric TSP noting some factors which influence the performance of the tabu search method.

Sadiq[[12]] worked on traveling salesman problem using Genetic algorithm to optimize delivery routes. Their application was to a company whose delivery routes scheduling from a central warehouse that has one hundred packages to be delivered with five trucks optimally. They concluded that they genetic algorithm is a good alternative to the greedy algorithm. Alhamdy *et al*[[2]] used ant colony optimization (ACO) algorithm to solve traveling salesman problem. They also compared the approach with Tabular search, simulated Annealing and Genetic algorithm and observed that in larger scales, the ant colony algorithm is more efficient than other algorithm in solving traveling salesman problem. Moon *et al*[[8]] presented an efficient genetic algorithm for solving traveling salesman problem with procedure constraints. The main concept in their proposed algorithm was a topological sort (TS); defined as an ordering of vertices in a directed graph. It was also observed that the result of the numerical experiments show a superior performance compared to other algorithms and produces an optimal solution as well.

Fereidouni[[3]] used a fuzzy multi-objective linear programming (FMOLP) in solving travelling salesman problem (TSP). This problem was applied to an uncertain environment and a numerical example was used to ascertain the effectiveness of the proposed method and the results show that the FMOLP model achieves higher satisfaction degrees. MohdRazali[[7]] developed a new genetic algorithm procedure which takes care of chromosome's repairing strategy based on topological sort, and also to generate only feasible solution during the evolutionary process. The new developed method was compared to the existing ones and was observed that this new method by MohdRazali[[7]] was capable of tackling large size problems and reach for optimal solutions. Stencek[[13]] developed an algorithm and compared his solution with the solution of four different algorithms. He also developed a software using Java programming language to solve traveling salesman problem.

Rostamiet *al*[[11]] modified the Gravitational Emulation Local Search (GELS) algorithm to solve the symmetric MTSP. The performance of the modified GELS was compared with other well-known optimization algorithm. Also both the efficiency and superiority of the modified GELS were compared based on the computed total travelled distance and time required to solve the modified traveling salesman problem. Akandwanahoet *al*[[1]] used a Dynamic Gaussian Regression method to solve a dynamic traveling salesman problem. They applied a non-stationary covariance function in GPR for the optimal tour of 22 city data set. When their results were compared to other existing approaches, it was found out that their proposed approach demonstrated superiority in obtaining good traveling salesman problem tour with less computational time in non-stationary conditions.

Filip & Otakar[[4]] reviewed the some method of solving traveling salesman problem and its application in logistic practice providing task solutions in a distribution company within specific conditions and other requirements of the transport management of that company on Czech Republic. The problem encountered by that company was how to optimally distribute chemical product to about 20 customers in 16 cities. Ochiai&Kanoh[[9]] provided a solution to real world delivery problems for home delivery services where there exist a large number of roads in a city. Their approach was a combination of ant colony optimization with Dijkstra algorithm under a hybrid meta-heuristics. The experimental result of their work showed that the proposed method is effective in a wide area road network.

Wang & Regan [[14]] worked on assignment models for local truck load trucking problems with stochastic service time and time window constraint. They developed an integer model for minimizing the total cost of transportation with fixed fleet size. And also they were able to present a solution to a multiple traveling salesman problem (MTSP) with time windows.

III. Methodology

In this work, the metric TSP, also known as delta-TSP or Δ -TSP was employed, the intercity distance satisfies the triangular inequality. A very natural restriction of the TSP is to require that the distance between cities form a metric to satisfy the triangular inequality, that is the direct connection from A to B is never farther than the route via intermediate C. Mathematically, $d_{AB} < d_{AC} + d_{BC}$.

The metric TSP satisfies the triangular inequality. In the metric TSP, the graph which contains all the nodes or locations to be visited by the salesman is been transformed into a metric form. The metric TSP collects the corresponding distance, time or cost of all the routes between each node in the graph and puts it in a matrix form. The metric TSP deals mostly with symmetric traveling salesman problem. While computing it in the

4	-	-	0	-	20	0	-	-	-	-	-	-	-	-	-	-
5	-	-	-	0	-	30	30	-	-	-	-	-	-	-	-	-
6	-	-	60	10	60	-	10	-	-	10	-	0	-	-	-	-
7	-	-	-	-	50	0	-	0	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	0	-	-	0	-	-	-	-	-	-
9	-	-	0	-	-	-	-	-	-	-	-	0	-	-	-	-
10	-	-	-	-	-	0	-	0	-	-	50	-	-	-	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0	-	-	-	50
12	-	-	-	-	-	0	-	-	10	-	10	-	60	-	60	-
13	-	-	-	-	-	-	-	-	-	-	-	50	-	0	50	-
14	0	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-
15	-	-	-	-	-	-	-	-	-	-	-	50	50	-	-	0
16	-	-	-	-	-	-	-	-	-	50	-	-	-	-	0	-

Table 4: Column minimization column 5 and 11

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	-	0	-	-	-	-	-	-	-	-	-	-	-	0	-	-
2	0	-	0	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	0	-	0	-	50	-	-	0	-	-	-	-	-	-	-
4	-	-	0	-	0	0	-	-	-	-	-	-	-	-	-	-
5	-	-	-	0	-	30	30	-	-	-	-	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0	-	-	-	-
7	-	-	-	-	30	0	-	0	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	0	-	-	0	-	-	-	-	-	-
9	-	-	0	-	-	-	-	-	-	-	-	0	-	-	-	-
10	-	-	-	-	-	0	-	0	-	-	40	-	-	-	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0	-	-	-	50
12	-	-	-	-	-	0	-	-	10	-	0	-	60	-	60	-
13	-	-	-	-	-	-	-	-	-	-	-	50	-	0	50	-
14	0	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-
15	-	-	-	-	-	-	-	-	-	-	-	50	50	-	-	0
16	-	-	-	-	-	-	-	-	-	-	40	-	-	-	0	-

Table 5:Penalty

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	0	-	-
2	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	-	-
3	-	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-	-	-	-	-
4	-	-	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-	-	-	-	-	-
5	-	-	-	0 ³⁰	-	30	30	-	-	-	-	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0 ¹⁰	-	-	-	-
7	-	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-	-	-	-	-
9	-	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰	-	-	-	-
10	-	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-	-	-	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0 ⁵⁰	-	-	-	50
12	-	-	-	-	-	0 ⁰	-	-	10	-	0 ⁴⁰	-	60	-	60	-
13	-	-	-	-	-	-	-	-	-	-	-	50	-	0 ⁵⁰	50	-
14	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	0 ⁵⁰	-	-	-
15	-	-	-	-	-	-	-	-	-	-	-	50	50	-	-	0 ¹⁰⁰
16	-	-	-	-	-	-	-	-	-	40	-	-	-	-	0 ⁹⁰	-

The assignment is from city 15 to city 16 (15 → 16)

Reduced matrix

Table 6:Row minimization for row 16 and column minimization for column 15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	0	-
2	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	-
3	-	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-	-	-	-
4	-	-	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-	-	-	-	-
5	-	-	-	0 ³⁰	-	30	30	-	-	-	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0 ¹⁰	-	-	-
7	-	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-
8	-	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-	-	-	-
9	-	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰	-	-	-
10	-	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-	-	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0 ⁵⁰	-	-	-
12	-	-	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-	60	-	10
13	-	-	-	-	-	-	-	-	-	-	-	50	-	0 ⁰	0 ¹⁰

14	0 ⁰	-	-	-	-	-	-	-	-	-	-	-	0 ⁶⁰	-	-
16	-	-	-	-	-	-	-	-	-	-	0 ⁰	-	-	-	-

The second assignment is from city 14 to city 13 (14 → 13)

Table 7:Reduced matrix

	1	2	3	4	5	6	7	8	9	10	11	12	14	15
1	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	0	-
2	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	-
3	-	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-	-	-
4	-	-	-	0 ³⁰	0 ⁰	-	-	-	-	-	-	-	-	-
5	-	-	-	0 ³⁰	-	30	30	-	-	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0 ¹⁰	-	-
7	-	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-	-	-
8	-	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-	-	-
9	-	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰	-	-
10	-	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0 ³⁰	-	-
12	-	-	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-	-	10
13	-	-	-	-	-	-	-	-	-	-	-	50	-	0 ⁶⁰
16	-	-	-	-	-	-	-	-	-	-	0 ⁰	-	-	-

The next assignment is from city 13 to city 15 (13 → 15)

Since city 14 has only one route, which is city one, city 14 is assigned to city 1

Table 8:Reduced matrix

	1	2	3	4	5	6	7	8	9	10	11	12	14
1	-	0 ⁰	-	-	-	-	-	-	-	-	-	-	0 ⁰
2	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-	-	-
3	-	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-	-
4	-	-	-	0 ³⁰	0 ⁰	-	-	-	-	-	-	-	-
5	-	-	-	0 ³⁰	-	30	30	-	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0 ¹⁰	-
7	-	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-	-
8	-	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-	-
9	-	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰	-
10	-	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-	-
11	-	-	-	-	-	-	-	-	-	50	-	0 ³⁰	-
12	-	-	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-	-
16	-	-	-	-	-	-	-	-	-	-	0 ⁰	-	-

The route is from city 1 to city 14 (1 → 14)

Since city 2 has only one route, which is city 1, city 2 is assigned to city 1

Table 9:Reduced matrix

	1	2	3	4	5	6	7	8	9	10	11	12
2	0 ⁰	-	0 ⁰	-	-	-	-	-	-	-	-	-
3	-	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-
4	-	-	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-	-
5	-	-	-	0 ³⁰	-	30	30	-	-	-	-	-
6	-	-	60	10	40	-	10	-	-	10	-	0 ¹⁰
7	-	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-
8	-	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-
9	-	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰
10	-	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-
11	-	-	-	-	-	-	-	-	-	50	-	0 ³⁰
12	-	-	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-
16	-	-	-	-	-	-	-	-	-	-	0 ⁰	-

The assignment is from city 2 to city 1 (2 → 1)

Since city 2 has only one route, which is city 3, city 3 is assigned to city 2

Table 10:Reduced matrix

s	2	3	4	5	6	7	8	9	10	11	12
3	0 ⁰	-	0 ⁰	-	50	-	-	0 ¹⁰	-	-	-
4	-	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-	-
5	-	-	0 ³⁰	-	30	30	-	-	-	-	-
6	-	60	10	40	-	10	-	-	10	-	0 ¹⁰
7	-	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-
8	-	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-
9	-	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰

10	-	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-
11	-	-	-	-	-	-	-	-	50	-	0 ⁵⁰
12	-	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-
16	-	-	-	-	-	-	-	-	-	0 ⁰	-

The assignment is from city 3 to city 2
 Since city 9 has only one route, which is city 12, city 12 is assigned to city 9

Table 11: Reduced matrix

	3	4	5	6	7	8	9	10	11	12
4	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-	-
5	-	0 ³⁰	-	30	30	-	-	-	-	-
6	60	10	40	-	10	-	-	10	-	0 ¹⁰
7	-	-	30	0 ⁰	-	0 ⁰	-	-	-	-
8	-	-	-	-	0 ¹⁰	-	-	0 ¹⁰	-	-
9	0 ⁰	-	-	-	-	-	-	-	-	0 ⁰
10	-	-	-	0 ⁰	-	0 ⁰	-	-	40	-
11	-	-	-	-	-	-	-	50	-	0 ⁵⁰
12	-	-	-	0 ⁰	-	-	10	-	0 ⁰	-
16	-	-	-	-	-	-	-	-	0 ⁰	-

The assignment is from city 12 to city 9 (12→9)
 Since city 16 has only one route, which is city 11, city 16 is assigned to city 11

Table 12: Reduced matrix

	3	4	5	6	7	8	10	11	12
4	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-	-
5	-	0 ³⁰	-	30	30	-	-	-	-
6	60	10	40	-	10	-	10	-	0 ¹⁰
7	-	-	30	0 ⁰	-	0 ⁰	-	-	-
8	-	-	-	-	0 ¹⁰	-	0 ¹⁰	-	-
9	0 ⁰	-	-	-	-	-	-	-	0 ⁰
10	-	-	-	0 ⁰	-	0 ⁰	-	40	-
11	-	-	-	-	-	-	50	-	0 ⁵⁰
16	-	-	-	-	-	-	-	0 ⁰	-

The assignment is from city 16 to city 11
 Since city 9 has only one route, which is city 3, city 9 is assigned to city 3

Table 13: Reduced matrix

	3	4	5	6	7	8	10	12
4	0 ⁰	-	0 ³⁰	0 ⁰	-	-	-	-
5	-	0 ³⁰	-	30	30	-	-	-
6	60	10	40	-	10	-	10	0 ¹⁰
7	-	-	30	0 ⁰	-	0 ⁰	-	-
8	-	-	-	-	0 ¹⁰	-	0 ¹⁰	-
9	0 ⁰	-	-	-	-	-	-	-
10	-	-	-	0 ⁰	-	0 ⁰	-	-
11	-	-	-	-	-	-	50	0 ⁵⁰

The assignment is from city 9 to city 3 (9→3)
 Since city 12 has only one route, which is city 6, city 6 is assigned to city 12

Table 14: Reduced matrix

	4	5	6	7	8	10	12
4	-	0 ³⁰	0 ⁰	-	-	-	-
5	0 ³⁰	-	30	30	-	-	-
6	10	40	-	10	-	10	0 ¹⁰
7	-	30	0 ⁰	-	0 ⁰	-	-
8	-	-	-	0 ¹⁰	-	0 ⁰	-
10	-	-	0 ⁰	-	0 ⁰	-	-
11	-	-	-	-	-	0	-

The assignment is from city 6 to city 12 (6→12)
 Since city 11 has only one route, which is city 10, city 11 is assigned to city 10

Table 15: Reduced matrix

	4	5	6	7	8	10
4	-	0 ³⁰	0 ⁰	-	-	-
5	0 ³⁰	-	30	30	-	-

7	-	30	0 ⁰	-	0 ⁰	-
8	-	-	-	0 ¹⁰	-	0 ⁰
10	-	-	0 ⁰	-	0 ⁰	-
11	-	-	-	-	-	0

The assignment is from city 11 to city 10 (11 → 10)
 Since city 8 has only one route, which is city 7, city 8 is assigned to city 7

Table 16:Reduced matrix

	4	5	6	7	8
4	-	0 ³⁰	0 ⁰	-	-
5	0 ³⁰	-	30	30	-
7	-	30	0 ⁰	-	0 ⁰
8	-	-	-	0 ¹⁰	-
10	-	-	0 ⁰	-	0 ⁰

The assignment is from city 8 to city 7 (8 → 7)
 Since city 8 has only one route, which is city 10, city 10 is assigned to city 8

Table 17:Reduced matrix

	4	5	6	8
4	-	0 ³⁰	0 ⁰	-
5	0 ³⁰	-	30	-
7	-	30	0 ⁰	-
10	-	-	0 ⁰	0 ⁰

The assignment is from city 10 to city 8
 Since city 5 has only one route, which is city 4, city 5 is assigned to city 4

Table 18:Reduced matrix

	4	5	6
4	-	0 ³⁰	0 ⁰
5	0 ³⁰	-	30
7	-	30	0 ⁰

The assignment is from city 5 to city 4 (5 → 4)

Table 19:Reduced matrix

	5	6
4	-	0 ⁰
7	30	0 ⁰

From the above table, the assignment is from city 4 to city 6 and from city 7 to city 5.

The best possible routes are from city 15 to city 16, city 14 to 13, city 1 to city 14, city 2 to city 1, city 3 to city 2, city 12 to city 9, city 16 to city 11, city 9 to city 3, city 6 to city 12, city 11 to city 10, city 8 to city 7, city 10 to city 8, city 5 to city 4, city 7 to city 5 and finally from city 4 to city 6.

Therefore desired route is given as:

Choba → Mbuogba → Location → Mile III → Mile I → Rumuola → Artillery → Rumuokurusi → Tank → Eneka → Eliozu → Rumuodara → Rumuokwuta → Rumuigbo → Rumuokoro → Rumuosi → Choba.

And the cost of the route is given as

$$50+50+100+50+100+100+50+50+100+70+50+40+50+50+50+50= \text{N}1010$$

The above solution is the analytical solution to the problem. Using the software “TSP SOLVER AND GENERATOR”, from the result given by the software, the optimum routes are:

Resulting path:

City 1 → City 14 → City 13 → City 15 → City 16 → City 11 → City 10 → City 8 → City 7 → City 5 → City 4 → City 6 → City 12 → City 9 → City 3 → City 2 → City 1.

The price is **1010** units

Which is the same as

Choba → Mbuogba → Location → Mile III → Mile I → Rumuola → Artillery → Rumuokurusi → Tank → Eneka → Eliozu → Rumuodara → Rumuokwuta → Rumuigbo → Rumuokoro → Rumuosi → Choba.

And this is the same route gotten from the analytical solution. The details of the software solution is given in Appendix.

V. Conclusion

The aim and objective of this project is to minimize the cost of distribution of product (water) in Junac table Water Company to various locations in Port Harcourt. In other to achieve this, the matrix method of solving TSP was adopted and also a software (TSP solver and Generator, v0.1) was used to also to determine the best (cheapest) route to be taken during the distribution of the product.

The best route gotten from the analytical solution (matrix method) has a minimum cost of ₦1010, also the best route gotten from the TSP solver and generator has a minimum cost of ₦1010.

It can be concluded that the company would make a higher profit if they can adhere to the above method gave the minimum cost of the route to be taken by one of their trucks. This method will yield maximum profit in the area of distribution if it is applied to all their trucks and their corresponding route.

References

[1] Akandwanaho, S. M., Adewumi, A. O., &Adebiyi, A. A. (2014). Solving Dynamic Traveling Salesman Problem Using Dynamic Gaussian Process Regression. *Journal of Applied Mathematics*, 2014.

[2] Alhamdy, S., Ahmad, S., Noudehi, A. N., &Majdara, M. (2012). Solving traveling salesman problem (TSP) using ants colony (ACO) algorithm and comparing with tabu search, simulated annealing and genetic algorithm. *J. Appl. Sci. Res*, 8(1), 434-440.

[3] Fereidouni, S. (2011). Solving traveling salesman problem by using a fuzzy multi-objective linear programming. *African Journal of mathematics and computer science research*, 4(11), 339-249.

[4] Filip, E., &Otakar, M. (2011). The travelling salesman problem and its application in logistic practice. *WSEAS Transactions on Business and Economics*, 8(4), 163-173.

[5] Knox, J. (1994). Tabu search performance on the symmetric traveling salesman problem. *Computers operations research*, 21(8), 867-876.

[6] Lin, S., & Kernighan, B. W. (1971). An effective heuristic algorithm for the traveling salesman problem.

[7] MohdRazali, N. (2014). *Genetic algorithm for process sequencing modelled as the travelling salesman problem with precedence constraints* (Doctoral dissertation, Dublin City University).

[8] Moon, C., Kim, J., Choi, G., &Seo, Y. (2002). An efficient genetic algorithm for the traveling salesman problem with precedence constraints. *European Journal of Operational Research*, 140(3), 606-617.

[9] Ochiai, J., &Kano, H. (2014). SOLVING REAL-WORLD DELIVERY PROBLEM USING IMPROVED MAX-MIN ANT SYSTEM WITH LOCAL OPTIMAL SOLUTIONS IN WIDE AREA ROAD NETWORK. *International Journal of Artificial Intelligence & Applications*, 5(3), 21.

[10] Reinelt, G. (1991). TSPLIB- a traveling salesman problem library. *Operations research society of America journal on computing*, 3(4),

[11] Rostami, A. S., Mohanna, F., Keshavarz, H., &Hosseinabadi, A. A. R. (2015). Solving Multiple Traveling Salesman Problem using the Gravitational Emulation Local Search Algorithm. *Applied Mathematics & Information Sciences*, 9(2), 699.

[12] Sadiq, S. (2012). The traveling salesman problem: optimizing delivery routes using genetic algorithms. In *SAS Global Forum, paper* (Vol. 161).

[13] Stencek, J. (2013). Traveling salesman problem.

[14] Wang, X., & Regan, A. (2001). Assignment models for local truckload trucking problems with stochastic service times and time window constraints. *Transportation Research Record: Journal of the Transportation Research Board*, (1771), 61-68.

APPENDIX A.
Variant #1 Task

Task:

---	50	---	---	---	---	---	---	---	---	---	---	---	50	---	---
50	---	50	---	---	---	---	---	---	---	---	---	---	---	---	---
---	50	---	50	---	100	---	---	50	---	---	---	---	---	---	---
---	---	50	---	70	50	---	---	---	---	---	---	---	---	---	---
---	---	---	70	---	100	100	---	---	---	---	---	---	---	---	---
---	---	100	50	100	---	50	---	---	50	---	40	---	---	---	---
---	---	---	---	100	50	---	50	---	---	---	---	---	---	---	---
---	---	---	---	---	---	50	---	---	50	---	---	---	---	---	---
---	---	50	---	---	---	---	---	---	---	---	50	---	---	---	---
---	---	---	---	---	50	---	50	---	---	100	---	---	---	---	---
---	---	---	---	---	---	---	---	---	100	---	50	---	---	---	100
---	---	---	---	---	40	---	---	50	---	50	---	100	---	100	---
---	---	---	---	---	---	---	---	---	---	---	100	---	50	100	---
50	---	---	---	---	---	---	---	---	---	---	---	50	---	---	---
---	---	---	---	---	---	---	---	---	---	---	100	100	---	---	50
---	---	---	---	---	---	---	---	---	---	100	---	---	---	50	---

Variant #1 Solution

Step #1

Selected route with (15;1Q126) part.

Step #2

Selected route with (14; 13) part.

Step #3

Selected route with (1; 14) part.

2 alternate candidates for branching: (2; 1), (16; 11).

Step #4

Selected route with (2;1) part.

2 alternate candidates for branching: (3;2), (16;11).

Step #5

Selected route with (3;2) part.

1 alternate candidate for branching: (16;11).

Step #6

Selected route with (12;9) part.

1 alternate candidate for branching: (16;11).

Step #7

Selected route with (9;3) part.

2 alternate candidates for branching: (13;15), (16;11).

Step #8

Selected route with (13;15) part.

1 alternate candidate for branching: (16;11).

Step #9

Selected route with (16;11) part.

Step #10

Selected route with (6;12) part.

1 alternate candidate for branching: (11;10).

Step #11

Selected route with (5;4) part.

1 alternate candidate for branching: (11;10).

Step #12

Selected route with (4;6) part.

2 alternate candidates for branching: (8;7), (11;10).

Step #13

Selected route with (7;5) part.

3 alternate candidates for branching: (8;7), (10;8), (11;10).

Step #14

Selected route with (10;8) part.

Step #15

Selected route with (8;7) part.

1 alternate candidate for branching: (11;10).

Resulting path:

City 1 → City 14 → City 13 → City 15 → City 16 → City 11 → City 10 → City 8 → City 7 → City 5 → City 4 → City 6 → City 12 → City 9 → City 3 → City 2 → City 1

The price is **1010** units.