Applying Tabu Search Algorithm in Public Transport: A Case Study for University Students in Mauritius

J. Cheeneebash and S. Jugee

***Abstract*—**In this paper, the Tabu search algorithm is used to solve a transportation problem which consists of determining the shortest routes with the appropriate vehicle capacity to facilitate the travel of the students attending the University of Mauritius. The aim of this work is to minimize the total cost of the distance travelled by the vehicles in serving all the customers. An initial solution is obtained by the TOUR algorithm which basically constructs a giant tour containing all the customers and partitions it in an optimal way so as to produce a set of feasible routes. The Tabu search algorithm then makes use of a search procedure, a swapping procedure and the intensification and diversification mechanism to find the best set of feasible routes.

***Keywords*—**Tabu Search, Vehicle Routing, Transport.

I. INTRODUCTION

MAURITIUS is a small island situated in the Indian Ocean which has a highly dense population of around 1.4 million. The mode of transport over the island is mainly by car and buses. The public transport is covered throughout the island by bus and taxis and there exists good road structures everywhere. The government provides free transport facilities for students to travel to their destination. But during peak time the route going towards the capital, Port- Louis is highly congested and the case is worse if there is an accident on the highway. The steady rise in internal road traffic has become a major problem with its impact also on land use, pollution and injury. Congestion is increasing the costs of doing business, as we waste hours in traffic (and hence wasted fuel) and production schedules are disrupted. Congestion is ultimately resulting in a less productive economy and reduced quality of life. Efficient decisions for managing the transportation of students can result in cost savings for the Government as well as preserve the environment. The transportation problem can be classified as a Mix Fleet Vehicle Routing Problem which involves a configuration of different types of vehicles and several customers to be satisfied and the aim would be to minimize cost of transportation. In this paper we have considered a heuristic method, a Tabu Search algorithm, to solve the travelling problem of a sample of students who travel from different parts of the island towards the University of Mauritius which is located in Reduit, in almost in the central part of the island.

J. Cheeneebash is with the Department of Mathematics, University of Mauritius, Mauritius (+230 403 7496); fax: +230 4656928; e-mail: jayrani@uom.ac.mu).

S. Jugee was with the University of Mauritius, Mauritius.

The paper is organized as follows: in section II we describe the Mix Fleet Vehicle problem followed by the Tabu search algorithm in section III. In section IV we give the simulation results and finally conclude in section V.

II. THE MIXED FLEET VEHICLE PROBLEM

The mix fleet vehicle problem can be mathematically described as follows [1]. Let G = (V, A) be an undirected connected graph, with $V = \{v_1, v_2, \ldots, v_n\}$, the set of nodes representing all the customers positions, and $A = \{ (v_i,v_j): v_i, v_j \in V\}$ forms the arcs which maintain the order of the customers, $i = 1, 2, 3, \ldots, n$, where n is the number of customers. Customers demand is denoted by q_i and a distance d_{ij} ($d_{ii} = 0$ for each $v_i \in V$}) is associated by each arc $(v_i,v_j) \in A$. There is a fleet t of different types of vehicles located at the depot. A capacity Q_k, a fixed cost F_k which is incurred by simply using a vehicle, and a variable cost c_k is associated to each type of vehicle k ($k = 1, 2, \ldots, f$). We assume that $Q_1 < Q_2 < \ldots < Q_f$ and $F_1 < F_2 < \ldots < F_t$. The travelling cost of each arc $(v_i, v_j) \in$ A by a vehicle of type k is $c_{ij} = c_k d_{ij}$. The problem consists of defining s set of routes and vehicles assigned to them so that the following constraints are considered: (i) satisfy customers' demand; (ii) visit each customer exactly once; (iii) a vehicle route starts and finishes at the depot and the capacity of the vehicle is not exceeded. The objective is to minimize the sum of fixed and variable costs of all the routes subject to constraints (i) – (iii). A mathematical formulation of the problem would be defined as follows. For any possible solution ε, the objective function is denoted by $f(\varepsilon)$. The objective function to be minimized is given by the equation:

$$f(\varepsilon) = \sum_{i=1}^{r} (F_i + c_i),$$

where r is the total number of routes in S, F_i is the fixed cost of the vehicle assigned to route i, c_i is the variable cost of route i.

III. TABU SEARCH ALGORITHM

Tabu search is an iterative search method that can be used to solve complicated optimization problems. The technique, introduced by Fred Glover [2] makes use of local search algorithm at each step of the iterations to generate the best solution in some neighbourhood. It is a procedure to explore the solution space beyond optimality and exploits the past history of the search in order to influence its future steps. A distinguishing feature of tabu search is embodied in its exploitation of adaptive forms of memory, which equips it to penetrate complexities that often confound alternative approach.

At each move, the algorithm seeks for the best progressing solution. Tabu search uses a tabu list which considers the moves used to obtain the best solutions at each step of the iterative method. The important factors in the Tabu search algorithm are local search method, neighbourhood structure, tabu conditions, aspiring conditions, stopping rule.

a. Initial Solution

Initially a giant tour is constructed including all the customers of the problem using the approach in [3]. We define a tour $G = (V, A)$ with $V = \{v_1, v_2, \ldots, v_n\}$, the set of nodes representing all the customers positions in the tour, and $A = \{ (v_i,v_j): v_i, v_j \in V\}$ forms the arcs which maintain the order of the customers, together with a travel cost c_{ij}. In the first stage, a set of s vehicle routes is built up. This algorithm contributes to find the shortest path that can be achieved to satisfy the demand of customers. Define a path starting from the depot to the closest customer. This step is repeated at each node $v_i \in V$ where $i = 1, 2, 3, \ldots, n$ with n denoting the total number of customers to be serviced in the current problem, until the last customer is attained. The second stage consists of allocating a vehicle to each possible route after the shortest path problem has been solved. Described a supplementary graph $G_1 = (V_1, A_1)$ where G_1 is one of the partition of the giant tour T, $V_1 = \{v_1, v_2, \ldots, v_t\}$ where t is referring to a certain number of customers, $t < n$, consists of the nodes on one of the s vehicle routes, and $A_1 = \{(v_e,v_f): 1 \le e \le f \le t)\}$. A travel cost c_{ef} is then assigned to each edge x^k_{ef}, where $k = 1, 2, \ldots, m$. x^k_{ef} is 1 if vehicle k travels from e to f else x^k_{ef} is 0. A partition of the giant tour is defined for feasible set of routes. The tour starts with the depot and each time a new customer is inserted, following their order number. The giant tour is divided into partitions in an optimal approach to produce a set of possible routes. The partitioning process builds up an auxiliary graph G_1 where $G_1 = (V, A_1)$. The fundamental aim of this process is to find a shortest path between the first and last customer of a partitioned tour. Table I shows the algorithm that partitions the giant tour T.

ALGORITHM 1
TOUR ALGORITHM

Step 1: Construct a giant tour T, where $T = \{v_1, v_2, \ldots, v_n\}$
Step 2: **for** each arc $(i,j) \in G$, **do**
- (a) Calculate c_{ij} and d_i the travel cost and demand associated to the route.
- (b) Calculate F_k, the fixed cost of vehicle k.

end do
Step 3: Calculate the shortest path from node v_1 to node v_n of graph G.
Step 4: Repeat steps 2-3 for other possible tours.

The process ends when there are no more customers in the tour.

b. Neighbourhood Structure

The Tabu Search Algorithm is based on neighbourhood moves. $N(\varepsilon\,\varepsilon)$ being the neighbourhood solution $\varepsilon\,\varepsilon$, is defined to be the set of solutions that can be reached from the solution $\varepsilon\,\varepsilon$. When a search is applied to a solution $\varepsilon\,\varepsilon$, a new solution $\varepsilon'\,\varepsilon'$ is generated in the neighbourhood of $\varepsilon\,\varepsilon$. The solutions which crop up during the neighbourhood search are routes that are composed of all nodes in the graph. Referring to a certain solution, the neighbourhood can be classified as any other solution by a swap move between two nodes of route i and route j respectively. A swap move consists of exchanging two customers belonging to two different routes. A trial swap of a customer of route i with a customer of route j if and only if (i) at least one customer of route j belongs to the δ-neighbourhood of customer i: (ii) at least one of the routes i or j contain more than one customer. Consequently, there will be no subtours. The first and last node, being the depot node, stay fixed and are not used for the swapping step. After a number of iteration, say n, the best objective value (minimum cost) in a neighbourhood is chosen.

c. Admissibility of Moves

A conventional tabu list [4] is created to prevent a customer that has been moved to return to its original route for the next θ iterations where θ is the tabu tenure. The tabu tenure changes systematically during the search according to the evolution of the results. The tabu restriction may be overridden if the move produces a solution that is better than the ones found in the past. This is referred to as the aspiration criterion. In the tabu search, a trial move to solution S' is regarded as admissible:

(i) if it is not currently in the tabu list;

(ii) or if it is tabu and infeasible, but the value of $f(S')$ is lower than the value of the best infeasible solution yet found;

(iii) or if it is tabu and feasible, but the value of $f(S')$ is lower than the value of the best feasible solution yet found.

D. Intensification and Diversification

Intensification and diversification are important in the tabu search algorithm as they allow exploitation of the search space that has not been visited. In this way, one can ensure that the optimal solution is one the one that has been trapped in a subset of the search space. The diversification mechanism is done by having another list in parallel with the tabu list which contains all the nodes in the graph G that formed part of some accepted solutions. The intensification is achieved as follows. The whole neighbourhood of the current solution is thoroughly explored at iteration ξ of the tabu search and the best neighbourhood, involving non-tabu moves, is chosen. This selected neighbourhood is then used as a current solution for iteration $\xi + 1$. The best solution at a given iteration is picked if there does not exist any other improving solution. Thereafter, the search procedure continues by investigating all the neighbourhoods.

E: Overview of the Tabu Search

The proposed algorithm makes use of the TOUR algorithm to construct a tour in order to evaluate ϑ initial solution. In this paper $\vartheta = 6$ was chosen and initialize ε^{BEST}, the best known solution with respect to the objective function. Define λ_{max} to be the total number of iteration and perform η number of iterations in all. The best non-tabu feasible solution is evaluated while keeping track of the tabu elements during the search. The objective cost, f(ε) is evaluated with respect to the known new solution. If the best new solution ε' is feasible with F(ε') < F(ε^{BEST}), set ε^{BEST} = ε'. Apply the diversification mechanism to search for solutions in unexamined neighbourhoods. If λ= η, the algorithm is stopped.

ALGORITHM 2
TABU SEARCH ALGORITHM

Step 1: (Initialization)
Calculate an initial solution using the Tour Algorithm
Step 2: (Search Procedure)
(a) Begin with λ = 1and define λ_{max}
(b) Select best non-tabu feasible solution $\varepsilon' \in N(\in)$
(c) Keep tabu elements of the movement
(d) Keep the new solution cost f(ε') in an array
(e) Do $\varepsilon - \varepsilon'$ and λ=λ+1
Step 3: (Swapping Procedure)
(a) Calculate ε' from ε applying the swap procedure at each route of ε
(b) Update $\varepsilon - \varepsilon'$ and λ=λ+1
(c) If f(ε') < f(ε^{BEST}) then
ε^{BEST} = ε' , λ_{BEST} = λ and *over* = 0
else
over = *over* + 1
Step 4: (Diversification/Intensification Mechanism)
(a) Split each route of ε
(b) Allocate the cheapest vehicles to satisfy demand
(c) Define the new solution cost f(ε')
Step 5: (Stopping Rule)
if λ >λ_{max} then stop

IV. SIMULATIONS RESULTS AND DISCUSSION

The algorithm was applied in the local context, more particularly in managing the transportation of students of the University of Mauritius. The University of Mauritius has a student population of 6000. An efficient managing for the transport of student will result in cost saving as well as less pollution. A case study was made from a sample of 156 students coming to the university by bus from various part of the island. The distances are from the different part of the island to Reduit, where the university is situated is calculated using the Global Positioning System (GPS) scheme (Calculator 2009) which offers approximate distances in kilometers.

We describe the assumption that we have used in this case study. Firstly we assume that students will be picked up at known pick up points. The school buses differ in capacity and road constraints inflicted on school buses have to be taken into account for vehicle selection. In the following section, the assumptions of the problem are given.

Assumptions
(a) All courses at the university start at 9 a.m.
(b) No entry roads and speed zones are neglected.
(c) Students will be picked up at several pickup points and not at their home.
(d) A limited number of school buses with varying capacities will be available.
(e) Approximate distances will be taken into account and will be measured in kilometers.
(f) School buses will maintain their service only in the morning.
(g) Buses will be well serviced at the depot before starting their routes so as to avoid breakdowns.
(h) The cost of traveling is calculated on the amount of diesel used per km.
(i) Drivers should not wait for late students on pickup points to avoid delays.
(j) Buses will start at five respective depots, where the University of Mauritius is included as one of them.

The Data set:

TABLE I
VEHICLE CAPACITIES AND FIXED COSTS

Vehicles Type	Fixed Cost	Capacity
A	95	15
B	120	20
C	185	30
D	250	40
E	370	60

The different type of vehicles with their carrying capacity and fixed cost is shown in Table I. Table II shows the different depots and their starting time.

Simulations were done using the algorithms described in the previous section and the optimal solution is given in Table III.

TABLE II
DEPOTS AT DISPOSAL

Depot	Start Time
Riviere du Rempart	6:00 a.m
Souillac	6:00 a.m
La Tour Koenig	7:00 a.m
Forest Side	7:00 a.m
Reduit (UOM)	6:00 a.m

The cost of each route per day has been calculated and is depicted by $\cos t = \rho \times (\Phi + \Psi + \Pi + \Theta) + \omega$ where ρ = Distance in km, Φ= cost per km for tyres, Ψ= cost per km for maintenance, Π= cost per km for lubricants, Θ cost per km for fuel, ω = fixed cost of vehicle.

Fig. 1 Road Map of Mauritius

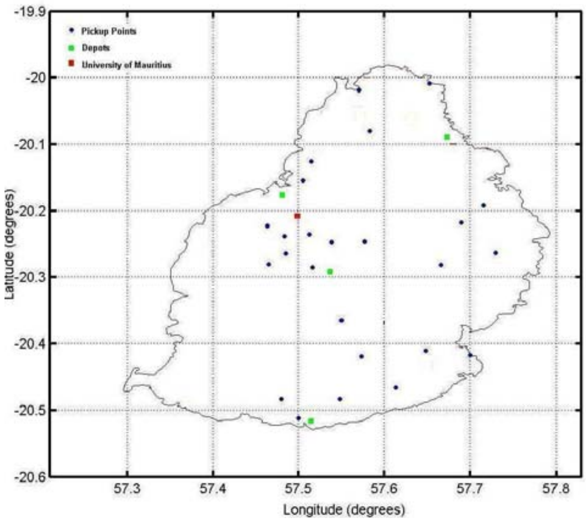

Fig. 2 Distribution of the depots and pick up points

TABLE III
OPTIMAL ROUTES TO THE UNIVERSITY

Route	Sequence	Vehicle Types	Distance (Km)	Number of students	Cost
1	UoM-Montagne Blanche-Bel Air-Centre de Flacq-Lallmatie-Quatier Militaire-St Pierre-Moka-UoM	C	71.37	30	1440
2	Riviere du Rempart-Grand Gaube-Grand Baie-Pamplemousses-Terre Rouge-St Croix UoM-Riviere du Rempart	C	82.51	22	1540
3	La Tour Koenig-Beau Bassin-Rose Hill-UoM-La Tour Koenig	C	18.07	30	475
4	Forest Side-Curepipe-Vacoas-Solferino-Quatre Bornes-UoM-Forest Side	E	29.9	59	935
5	Souillac-Chemin Grenier-Riviere des Anguilles-Grand Bois-Mahebourg - New Grove-Union Park-UoM-Souillac	A	120.09	15	2368
			321.94	156	6758

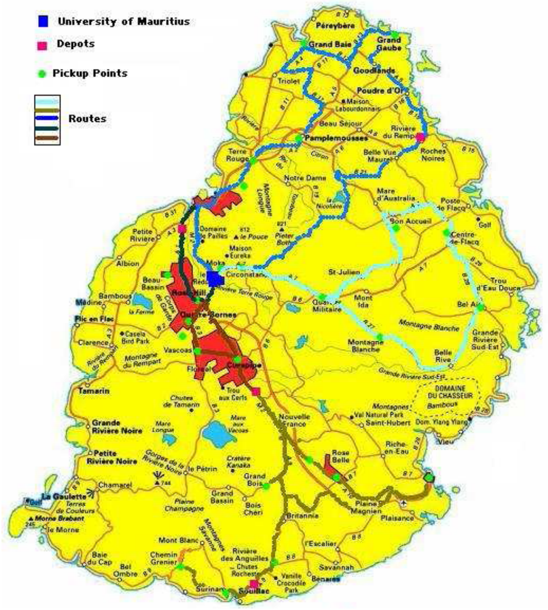

Fig. 3 Optimal Routes obtained with Tabu Search

V. Conclusion

In this paper, a tabu search algorithm is used to solve the Mix Fleet Vehicle Routing problem. The algorithm was implemented using Matlab 7.0 to find the shortest path by the TOUR algorithm which generates the initial solution and to find the best route compositions in order to minimize the total cost. The algorithm was tested on a real life situation and it was done on a sample of university students from different parts of the island. The solutions obtained gave an idea of the different allocation of bus capacities for the different routes, thus an important tool in decision making for the transportation management.

References

[1] B. E. Gillet, and L.R. Miller, "A heuristic algorithm for the vehicle dispatch problem", *Journal of Operations Research*, 1974, 22, pp. 340-349.

[2] F. Glover, and M. Laguna, "Tabu Search: Modern heuristic techniques for combinatorial problems", *R.C Reeves*, 1995, pp. 70-150.

[3] B. Golden et al, "The fleet size and mix vehicle routing problem", *Computers and Operations Research*, 1984, 11, pp. 49-66

[4] F.H. Liu, and S.Y. Shen, "The fleet size and mix vehicle routing problem with time windows", *Journal of the Operational Research Society*, 1999, 50, pp. 721- 732.