DEVELOPMENT OF A DECISION SUPPORT MODEL FOR OPTIMIZATION OF TOUR TIME TO VISIT TOURIST DESTINATION POINTS IN A CITY

Jagat Kumar Shrestha*

**Jagat Kumar Shrestha, Lecturer, Department of Civil Engineering, Pulchowk Campus, Institute of Engineering, Tribhuvan University, Kathmandu, Nepal,*

E-Mail: jagatshrestha@ioe.edu.np

ABSTRACT

Abstract-There may be a number of tourist destinations in a city; however, a tourist may have limited time that can be spent for tourism. The tourist may bee interested to visit as many tourist destinations as possible in the limited time. In this situation, the tourist want to optimize their travelling time and moments of leisure, taking the opportunity to visit the desired city attractions. This paper proposes a decision support model for optimization of tour time to visit tourist destination points in a city which deals with edge removal from a network and searching for alternate short routes to optimize the total tour time.

Keywords- *Optimization, network, tourist destination, tour time*

I. INTRODUCTION

Mostly tourists visit a city for a few days. It is not possible to visit every tourist destinations within a limited duration. Or, tourist may not be interested to visit all the destinations. Tourists want to use their free time in an optimal way [1], [2]. Usually, the tourist has a preference list (wish list) of desired Points of Interest (POI). Generally, this personal selection is based on available information found on web sites, in articles in magazines or in guidebooks. Once the selection is made, the tourist has to decide on a route, to the POIs considering the available time.

Web based decision support applications can be very useful aids for tourists for tour planning. Based on the selection of POIs an optimal route between them can be identified [3]. Travelling Salesperson Problem (TSP) [4] can be used as a starting point to plan tour trips [5]. A mobile tourist guide [3] use the Orienteering Problem (OP) [6] and its extensions to solve Tourist Trip Design Problems.

As the tourist is not visiting all the tourist destinations in the city network, the problem is not a TSP. A feasible solution of the TSP contains all the tourist destinations. However, the solution of this problem is a sub-tour in the original network of the city which contains POIs only; however, sub-tour is not a feasible solution in the TSP.

The OP is a combination of vertex selection and determining the shortest Hamiltonian path between the selected vertices. As a consequence, the OP can be seen as a combination between the Knapsack Problem and the TSP. The OP's goal is to maximise the total score collected, while the TSP tries to minimise the travel time or distance. Furthermore, not all vertices have to be visited in the OP. Determination of the shortest path between the selected vertices will be helpful to visit as many vertices as possible in the available time. The OP is the selective travelling salesperson problem [7], [8]. However, the tourist usually may start tour from nearby tourist attraction from the hotel where s/he stays and has to return back to the same location after completion of the tour. Furthermore, time spending at a vertex and the time to reach the vertex are independent and often contradictory to each other. This makes it difficult to select the vertices that will be part of the optimal solution. Therefore, heuristics may not efficiently explore the whole solution space. As the selected number of vertices in the network increases, the complexity of problem and solution time increase rapidly [9]. In this context, solution of the problem as the TSP is less complicated rather than the solution as the OP.

This paper proposes a decision support model that optimizes tourist tour time based on the selected tourist destinations (POIs) by the tourist in the original network. The problem is proposed to be solved reducing the tour problem from TSP in the original network to a reduced TSP in the reduced network. First, Section II presents modelling of the problem considering POIs. Next, Section III discusses the solution aspects of the problem. Section IV explains how the new model can be used to a tourist planning problem. Finally, Section V concludes the paper and points out some interesting research questions.

II. MODELLING THE PROBLEM

A tourist TSP is a mathematical optimisation problem that consists of a set of locations. The pair wise travel times between the locations are known. The goal is to find a tour that minimises the total length during visiting the tourist destinations. The total tour time (in route and time spent at POIs) cannot exceed the maximum amount of time the tourist has available.

Each tourist destination can be visited at most once. Hence, the problem can be a TSP consisting of the POIs for optimization of travel time in the network and time to be spent at POIs can be added to find the total tour time.

The problem has a network with a set of N vertices in a graph G = (V, A) where V= {$v_1, \ldots, v_N$} is the vertex set and A is the arc set. In this definition, the time to be spend T_i, is associated with each vertex $v_i \in V$ and the travel time t_{ij} is associated with each arc $a_{ij} \in A$. In this problem v_1 coincides with v_N. Using the notation introduced above, the problem is formulated as an integer problem. The following decision variables are used: $x_{ij} = 1$ if a visit to vertex i is followed by a visit to vertex j – 0 otherwise defined only for $i<j$.

For a symmetric TSP ($t_{ij} = t_{ji}$), the problem can be formulated as follows [4]:

Minimise:

$$z = \sum_{i}^{N} \sum_{j>i}^{N} t_{ij} x_{ij}$$

(1)

Subject to:

$$\sum_{j>i}^{N} x_{ji} + \sum_{j>i}^{N} t_{ij} x_{ij} = 2 \quad \forall i$$

(2)

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \quad \forall S \subset N$$

(3)

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

(4)

The objective function (1) is to minimise travelling time in the network. Constraints (2) ensure the connectivity of the path and guarantee that every vertex is visited at most once. Constraints (3) are necessary to prevent sub-tours. Constraints (4) show the binary integrality. This formulation have a symmetric travel times between the vertices ($t_{ij} = t_{ji}$). This corresponds to an undirected complete graph G.

Finally, the following equation (5) gives the total tour time which is minimum travel time plus time to be spent at each node i. This time is to be compared with time budget of the tourist ($T < T_{max}$), maximum time available for tourism.

$$T = z + \sum_{i}^{N} T_i \quad \forall i$$

(5)

III. SOLUTION APPROACH

A number of approaches offer solution of this type of problems. There are branch and bound and heuristic algorithms. Tour Planning in Mobile Tourism [10] uses a nearest neighbour approach, which iteratively adds the closest available visit to the tour. A dynamic tour guide search [11] uses a tree based search. Genetic algorithm can also be used to find near optimal solutions [12].

A city includes a number of tourist destinations connected by a transportation (road/rail) network. Considering the entire tourist destinations, we can form distance matrix in the transportation network. Commonly, shortest path algorithms such as Dijkstra [13] can be used to calculate distance between any two destinations in the network. Moreover, a short path matrix of the network can be found utilizing Floyd-Warshall algorithm [14] which gives the shortest distance to other destinations. We can mark POIs in the network. One way of planning a custom trip to a city is selecting hotel(s) and next filling the available time with POI visits in a nearest neighbour fashion, which may also indicate the nearby tourist destination as a starting destination of a tour.

From the short distance matrix, we can form a smaller network considering only POIs taking the edge as short distances between the POIs. In this way, the tour utilise the intermediate nodes and edges of the network through which it would be shorter to reach to desired destinations resulting to removal of some edges and nodes from the original network.

If the tourist visits all the tourist destination, the problem becomes a vertex weighted TSP. As the tourist usually have some POIs in the limited time budget, the network can be reduced to a smaller network. However, the entire network to be defined to minimize travelling distance between the POIs without considering the weight of non visiting nodes. Furthermore, the problem reduces to a smaller TSP which is a sub-tour within the original network. Then the problem can be solved as a TSP considering vertex weights using a standard algorithm such as Nearest Neighbour Algorithm. The complexity of the problem may be significantly reduced. The solution approach is implemented in the road network of Kathmandu city as a case study and presented in the following Section IV.

IV. APPLICATION OF THE MODEL IN KATHMANDU CITY NETWORK

Kathmandu, Nepal's capital is full of historical palaces and temples. Major POIs of tourist in Kathmandu city are Basantapur Durbar Square (with temples dating back to the 12th century), Boudhanath Stupa (a world heritage site), the Pashupatinath Temple (country's the most important Hindu temple, on the banks of the Bagmati river), the Royal Palace (the site of the infamous 2001 massacre of the Royal Family, and now converted into the Narayanhiti Palace Museum). The Swayambhunath Stupa (meaning the 'self-created' Stupa, aka the Monkey Temple on a hilltop to the west of Kathmandu), the Kopan Monastery (a gated community of Buddhist monks on a hilltop north of Boudhanath, the Royal Botanical Gardens (surrounded by an evergreen forest, are a site of outstanding beauty and the Garden of Dreams is a beautiful enclave in 5 minutes walking distance from the tourist centre of Thamel.

Kathmandu is also the gateway to the Bhaktapur Dubar Square and Patan Durbar Square.

All the tourist destinations lie in the city road network as shown in Figure 1. Each destination is represented by a unique node number. This network shows only the 12 major tourist destinations in the city. The distance between the tourist destinations are estimated as time required in minutes to cover the distance based on the data provided by tour operators in Kathmandu city and these distances are presented in Table 1 in the form of distance matrix.

The short path matrix (Table 2) shows only the travelling time required in route. A minimum time is necessary at each tourist destinations. However, the spending time at a tourist destination depends on her/his interest at the tourist destination. For this test instance, time estimated in each tourist destination is shown in Table 3.

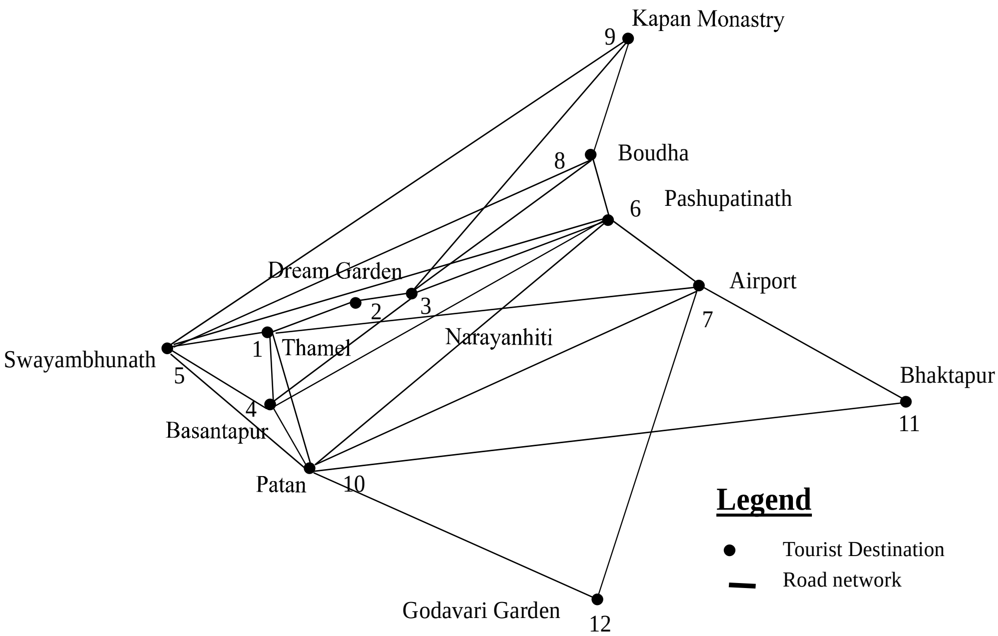


Figure 1: Major tourist destinations network in Kathmandu city

Table 1
Distance Matrix (time in minutes)

Destination	node	1	2	3	4	5	6	7	8	9	10	11	12
Thamel	1	0	5	0	10	15	0	35	0	0	45	60	0
Dream Garden	2	5	0	5	0	0	0	0	0	0	0	0	0
Narayanhiti	3	0	5	0	10	0	30	0	40	45	0	0	0
Basantapur	4	10	0	10	0	15	30	0	0	0	45	0	0
Swayambhunath	5	15	0	0	15	0	45	0	50	55	55	0	0
Pashupatinath	6	0	0	30	30	45	0	5	10	0	15	0	0
Airport	7	35	0	0	0	0	5	0	0	0	25	20	60
Boudha	8	0	0	40	0	50	10	0	0	10	0	0	0
Kapan Monastry	9	0	0	45	0	55	0	0	10	0	0	0	0
Patan	10	45	0	0	45	55	15	25	0	0	0	40	45
Bhaktapur	11	60	0	0	0	0	0	20	0	0	40	0	0
Godavari Garden	12	0	0	0	0	0	0	60	0	0	45	0	0

A short path matrix is calculated using Floyd-Warshall algorithm and presented in Table 2.

Table 2
Short Path Matrix (time in minutes)

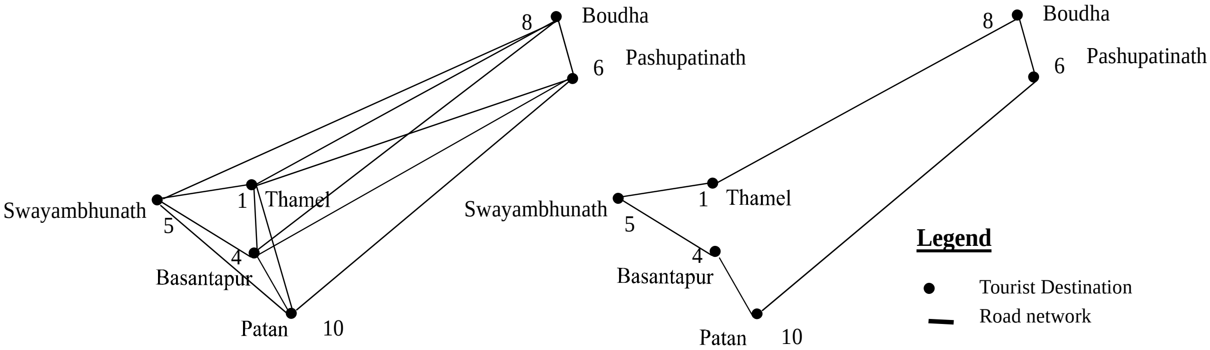
Destination	node	1	2	3	4	5	6	7	8	9	10	11	12
Thamel	1	0	5	10	10	15	40	35	50	55	45	55	90
Dream Garden	2	5	0	5	15	20	35	40	45	50	50	60	95
Narayanhiti	3	10	5	0	10	25	30	35	40	45	45	55	90
Basantapur	4	10	15	10	0	15	30	35	40	50	45	55	90
Swayambhunath	5	15	20	25	15	0	45	50	50	55	55	70	100
Pashupatinath	6	40	35	30	30	45	0	5	10	20	15	25	60
Airport	7	35	40	35	35	50	5	0	15	25	20	20	60
Boudha	8	50	45	40	40	50	10	15	0	10	25	35	70
Kapan Monastry	9	55	50	45	50	55	20	25	10	0	35	45	80
Patan	10	45	50	45	45	55	15	20	25	35	0	40	45
Bhaktapur	11	55	60	55	55	70	25	20	35	45	40	0	80
Godavari Garden	12	90	95	90	90	100	60	60	70	80	45	80	0

Table 3
Spending time at tourist destinations

Destination	Node	Minimum Time in minutes (T_i)	Range of spending time
Thamel	1	0	2-8 hours (1day)
Dream Garden	2	120	2-3 hours
Narayanhiti	3	120	
Basantapur	4	120	2-8 hours (1 day)
Swayambhunath	5	60	1-2 hours
Pashupatinath	6	120	2-8 hours (1 day)
Airport	7	-	
Boudha	8	60	1-2 hours
Kapan Monastry	9	120	2-8 hours (1 day)
Patan	10	120	2-8 hours (1 day)
Bhaktapur	11	120	2-8 hours (1 day)
Godavari Garden	12	180	3-8 hours (1 day)

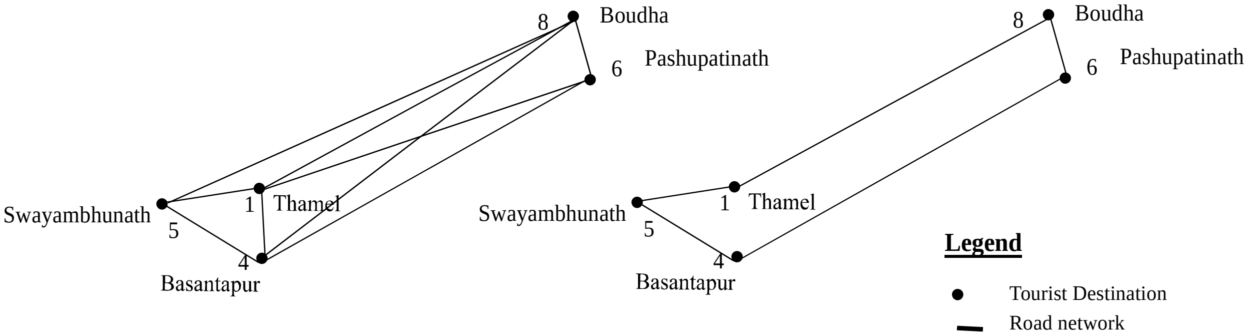
For this test instance of the model developed, let us assume a tourist has only a day (T_{max} = 8 hours) for tourism in Kathmandu city. Depending on her/his interest, s/he can design a trip in the city based on the information in Figure 1, Table 1, Table 2, and Table 3. For example, the tourist stays at Thamel and plans a tour her/his POIs are Basantapur, Swayambhunath, Pashupatinath, Patan and Bouda as shown in Figure 2, which is the reduced network from the original network (Figure 1) and time required for travelling is 95 minutes based on the solution of reduced TSP with nodes 1, 4, 5, 6, 8, and 10 and solution of this TSP is 1-8-6-10-4-5-1 (Figure 2(b)), using nearest neighbour algorithm. The time s/he has estimated to spend in the POIs is 480 minutes from Table 3. The total tour time is 575 minutes (T = 9 hours 35 minutes) which exceeds T_{max} , hence, is not be feasible and s/he may drop one POI (e.g. Patan, node 10). The network is further reduced because of removal of node 10 and edges 6-10 and 10-4 in the previous reduced network as shown in Figure 3. Then solving for the network, total tour time required will be 435 (75 + 360) minutes (T = 7 hours 15 minutes) with solution 1-8-6-4-5-1 (Figure 3(b)) which is a feasible solution in the time budget.

The network shown in Figure 2 is the reduced network from the original network which includes the POIs among the tourist destinations; however, the network inherits the properties of the original network. We can note that the distance between the POIs is the shortest distance from the original network. For example, distance from 1 to 6 in Figure 3(a) is 20 minutes which is sum of distance 1 to 2, 2 to 3, and 3 to 6 although node 2 and node 3 is not seen in the reduced network. In this way node 2 and node 3 and edges 1-2, 2-3 and 3-6 is removed and replaced by 1-6. The problem size is reduced. Then, we can use simply nearest neighbour algorithm to solve the problem as a TSP in which the tourist starts from POI 1, makes tour to all POIs and returns back to POI 1. For this, the solution is 1-8-6-4-5-1(Figure 3(b)). A tourist can hire a car or consult a tour operator and enjoy the trip with maximum utilization and optimization of precise time for leisure.



(a) network with POIs 1, 4, 5, 6, 8, 10 (b) Solurion 1-8-6-10-4-5-1

Figure 2: Reduced network with POIs 1, 4, 5, 6, 8, 10



(a) network with POIs 1, 4, 5, 6, 8 (b) Solution 1-8-6-4-5-1

Figure 3: Reduced network with POIs 1, 4, 5, 6, 8

This implementation of model to Kathmandu city shows that the model developed in this paper is applicable to solve tourist trip panning in a city. Also this model is applicable to urgent delivery of goods at different locations such as fuel.

V. CONCLUSIONS AND FUTURE WORKS

The proposed decision support model can integrate selection and routing of tourist destinations taking as a TSP. A big network can be reduced to a smaller network considering POIs which significantly reduces the complexity of the problem. Standard solution techniques such as nearest neighbour algorithm can be used as the removal of nodes and edges reduces the problem to a simple and handy. Hence, the model can be considered as a practical way of solution for tour time optimization in a city network.

Future work includes development of decision support model incorporating support for hotel selection and tour time optimization based on POIs in a city.

REFERENCES:

[1] R. Oppermann & M. Specht (1999). A nomadic information system for adaptive exhibition guidance. Archives & Museum Informatics, Vol. 13, Pp. 127-138.

[2] D.V. Keyson (2004). An electronic mobile guide for Artis zoo. Technical Report, Intelligence in Products Group, Faculty of Industrial Design, Delft University of Technology.

[3] P. Vansteenwegen & D. Van Oudheusden (2007). The Mobile Tourist Guide: an OR Opportunity. *OR Insight*, Vol. 20, No. 3, Pp. 21-27.

[4] Dantizig and B. George (1963). Linear Programming and Extensions, Princeton, *NJ: Princeton UP*, Sixth Printing, 1974, Pp. 545-547.

[5] J.M. Godart (2001). Combinatorial optimisation for trip planning. *Belgian Journal of Operations Research, Statistics and Computer Science*, Vol. 41 No. 1-2, Pp. 59-68.
[6] I. Chao, B. Golden & E. Wasil, (1996). Theory and methodology – the team orienteering problem. *European Journal of Operational Research* Vol. 88, Pp. 464– 474.
[7] G. Laporte & S. Martello (1990). The selective travelling salesman problem. *Discrete Applied Mathematics*, Vol. 26, Pp. 193–207.
[8] T. Thomadsen & T. Stidsen (2003). The quadratic selective travelling salesman problem. Informatics and Mathematical Modelling Technical Report 2003-17, *Technical University of Denmark*.
[9] D. Oudheusden (2008). A personalised tourist trip design algorithm for mobile tourist guides. *Applied Artificial Intelligence*, Vol. 22 No. 10, Pp. 964–985.
[10] C.C. Yu & H.P. Chang (2009). E-Commerce and Web Technologies, volume 5692 of LNCS, chapter Personalized Location-Based Recommendation Services for Tour Planning in Mobile Tourism Applications, Pp. 38-49, Springer Berlin / Heidelberg.
[11] K. T. Hagen, R. Kramer, M. Hermkes, B. Schumann & P. Mueller (2005). Semantic matching and heuristic search for a dynamic tour guide, Information and Communication Technologies in Tourism. Springer.
[12] C. Moon, J. Kim, G. Choi & Y. Seo (2002). An efficient genetic algorithm for the travelling salesmen problem with precedence constraints, *European Journal of Operational Research* Vol. 140, Pp. 606-617.
[13] G. Gallo & S. Pallattino (1986). Shortest path methods: A unified approach, *Mathematical Programming Study*, Vol. 26, No. 38, Pp. 64.
[14] R. W. Floyd (1962). Algorithm 97: Shortest Path, *Communications of the ACM*, Vol. 5, No. 6 Pp. 345.

Author Biography

Jagat Kumar Shrestha has received his PhD in 2013 in Civil Engineering with emphasis in Infrastructures Planning and Public Facility Locations from University of Aveiro, Portugal. He is graduated in 1993 in Civil Engineering form Institute of Engineering, Tribhuvan University, Nepal. He has master degree in Engineering, Economics and Management from Tribhuvan University. He is a faculty of Civil Engineering at Institute of Engineering, Tribhuvan University since 1998. He has experiences in the areas of planning, design and construction of engineering infrastructures. His interest is in Operations Research.