Journal of Computing Research in Innovation (JCRINN) Vol 1, No 1 (2016)
https://crinn.conferencehunter.com

Article 2

Travelling Choices Made Easier using Dijkstra Algorithm

Wardah Mohd Nor, Noraini Noordin
Faculty of Computer & Mathematical Sciences
Universiti Teknologi MARA Perlis Branch

Abstract
Current inflating high cost of living may affect mobility of students to commute between their university and any other destination. In an attempt to help solve or minimise the problem, several proposed travel alternative modes from this research may help students to identify the shortest path and minimum cost path in a journey. In finding the minimum total travel cost and shortest total completion time for a journey between two destinations, the step-by-step Dijkstra algorithm was applied in the time-dependent shortest path problem (TDSPP) and cost-dependent shortest path problem (CDSPP) in this research. Interestingly, both problems treated separately mapped out two different paths with two different costs. The former mapped out a route from UiTM Perlis Branch to Arau by taxi, train from Arau to Alor Setar, and Malaysia Airlines flights from Alor Setar to Kuala Lumpur and on to Johor Bahru. The other route was best described by taxi from Arau to Alor Setar, Air Asia flight from Alor Setar to Kuala Lumpur and train from Kuala Lumpur to Johor Bahru.

Keywords: *shortest path problem, dijkstra algorithm, network model, time-dependent SPP, cost-dependent SPP.*

Introduction

Economy of a country is very much affected by movement of goods and services through land, air and water (Nagurney, 2007). It is particularly important for students to move between two destinations. By constructing a network model to represent the journey between two destinations that involves multiple types of transportation, this study has been able to determine the shortest cost-efficient path between two destinations, in terms of time and money using Dijkstra algorithm In particular, this simple and well-known effective algorithm has been chosen to solve the time-dependent shortest path problem (TDSPP) and cost-dependent shortest path problem (CDSPP) in this research.

Literature Review

The models for both TDSPP and CDSPP were drawn separately.

i. Network Diagram

A network diagram consists of a set of nodes that are linked by arrows. It defines the relationship between nodes and edges, direction of edges, as well as cost of nodes and edges. It is denoted by (N, A), where N is a set of nodes and A is the set of edges (Taha, 2006). It is also an important mathematical program that has been applied in many fields like communication networks, social networks and scientific collaboration networks (Lloyd &Valeika, 2012). In addition, a network model can help to solve optimization problems such as transportation problem, critical path, shortest path, minimum cost flow problems, and many more. In this study, a network model has been used to determine the shortest path from UiTM Perlis Branch (UP) to Johor Bahru (JB) with minimum time and fare.

ii. The Shortest Path Problem (SPP)

According to Kamiński et al. (2011), SPP finds the path between two nodes in a graph that minimizes the sum of the weights of its constituent edges. For the current research, SPP has helped to find the best combination of transportation modes that would give shortest time and minimum cost between each pair of nodes. d_{ij}, travel time needed from node i to node $j, (i, j) \in A$ has been used to represent delay time. Delay time is time-dependent on the start time for the travel. In other words, the function $d_{ij}(t)$ returns time to travel from i to j when leaving i at time t, thus arrival time at node j is $t+d_{ij}(t)$ (Kamiński et al., 2011). With respect to the current research, the delay function has been defined for deterministic discrete time problems, where the time on each arc was known and finite with certainty.

iii. Dijkstra Algorithm

Dijkstra algorithm is especially effective to solve a single-source shortest path problem in a directed graph with nonnegative weight. It is considered to be a simpler but faster version of Ford's algorithm. However, weight functions needed to be identified ahead of time, are monotonic and do not change (Abbasi & Ebrahimnejad, 2011).

Dijkstra maintains an array of provisional distance, d for each node and size of the search space is $O(n^2)$ and $n/2$ nodes on the average. When all target nodes are reached, the path can be stopped (Murota & Shioura, 2014).

In this algorithm, set D for all extended nodes and set U for unintended nodes are fixed, set D will increase as the minimal weight node from U is updated and the procedures will continue until all nodes are in D (Jasika et al., 2012). Figure 1 displays the step-by-step application of this algorithm as adapted from Jasika et al. (2012) and is explained below:

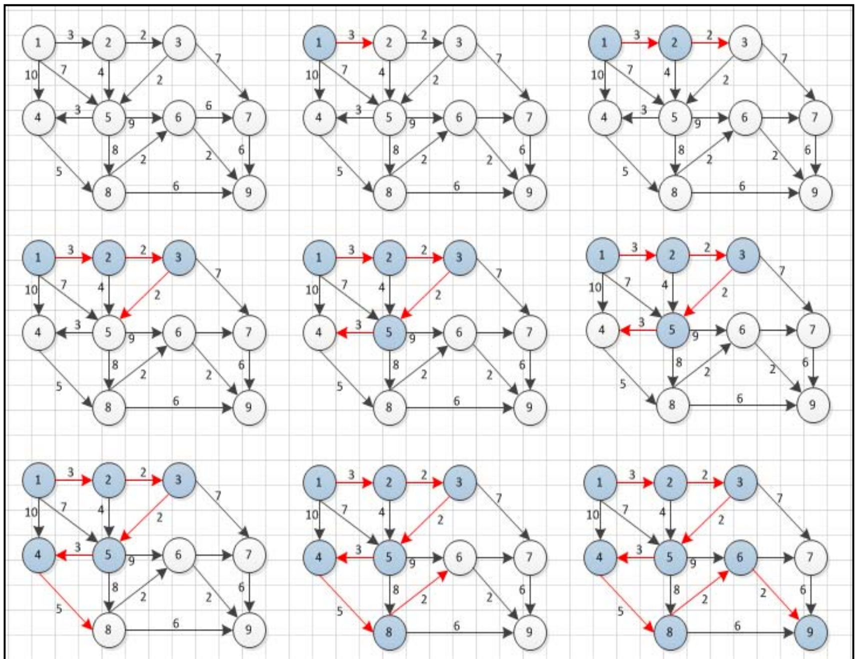

Figure 1: Dijkstra algorithm in steps

1. Assign zero weighted value to a source node, s which is tagged as permanent, (0, p). This now becomes the current node. Every other node is assigned a distance value of ∞ and is tagged as temporary (∞, t).

2. Letting i be the index of the current node, find another temporary tagged node that can be connected with i and update the distance values of (i, j) as $newd_j = \min(d_j, d_i + c_{ij})$, c_{ij} is the cost of the link (i, j).

3. Determine a node j with the smallest distance value d_j among all nodes connected with i, such that, $\min_j d_j = d_j^*$.

4. Change the label of node j^* to permanent and designate it as the current node.

5. By repeating all steps beginning step 2, the path has reached the end when all nodes have been permanently tagged.

Methodology

An analysis of the journey from UP to JB was done. It involved three common types of transport: bus, train, and airplanes. Selected transportation agencies were Transnasional, Sri Maju, City Express, AirAsia, Malaysia Airlines (MAS), Malindo Air, and Keretapi Tanah Melayu (KTM) Berhad.

i. Current Alternatives for Travel Routes out of UP

Firstly, to apply the Dijkstra algorithm, the assumptions made were i) UP was the start point, ii) JB was the terminal point, iii) varying price of flight tickets, and iv) mode of transport to the first activity port was taxi. Secondly, 245 selection of travel paths in the three-phase journey from UP to JB were constructed based on the condition that taxi was the mode of travels beginning from A (UP - Arau), C (UP - Alor Setar), and D (UP - Kangar).

ii. Network Model Construction

The first step in finding the shortest path is to determine activities that are related to the project (Marasovic and Marasovic, 2006), the first step to find the shortest path in a journey is to list the activities involved. Based on 21 activities, a precedence table as shown in Table 1 was drawn.

Table 1: Precedence Table

Task	Precedence	Description	Task	Precedence	Description
A	-	UP - Arau by taxi	**S**	A,B,C,D,E,F,G,H,I	Alor Setar - KL by MAS
B	A	Arau - Alor Setar by KTM	**T**	S	Dummies
C	-	UP - Alor Star by taxi	**U**	A,B,C,D,E,F,G,H,I	Alor Setar - KL by Malindo Air
D	-	UP - Kangar by taxi	**V**	U	Dummies
E	D	Kangar - Alor Setar by Sri Maju	**W**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by Sri Maju
F	D	Kangar – Alor Setar by Transnasional	**X**	W	Dummies
G	F	Dummies	**Y**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by Transnasional
H	D	Kangar – Alor Setar by City Express	**Z**	Y	Dummies
I	H	Dummies	**AA**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by City Express
J	A,B,C,D,E,F,G,H,I	Alor Setar - KL by Sri Maju	**AB**	AA	Dummies
K	J	Dummies	**AC**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by KTM
L	A,B,C,D,E,F,G,H,I	Alor Setar - KL by Transnasional	**AD**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by Air Asia
M	L	Dummies	**AE**	AD	Dummies
N	A,B,C,D,E,F,G,H,I	Alor Setar - KL by City Express	**AF**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by MAS
O	N	Dummies	**AG**	AG	Dummies
P	A,B,C,D,E,F,G,H,I	Alor Setar - KL by KTM	**AH**	J,K,L,M,N,O,P,Q,R ,S,T,U,V	KL - JB by Malindo Air
Q	A,B,C,D,E,F,G,H,I	Alor Setar - KL by Air Asia	**AI**	AH	Dummies
R	Q	Dummies			

The activity-on-arc techniques were used to construct two models for the time-dependent shortest path problem (TDSPP) and cost-dependent shortest path problem (CDSPP). The network for CDSPP is shown in Figure 2. The weights were given in terms of money in the TDSPP network.

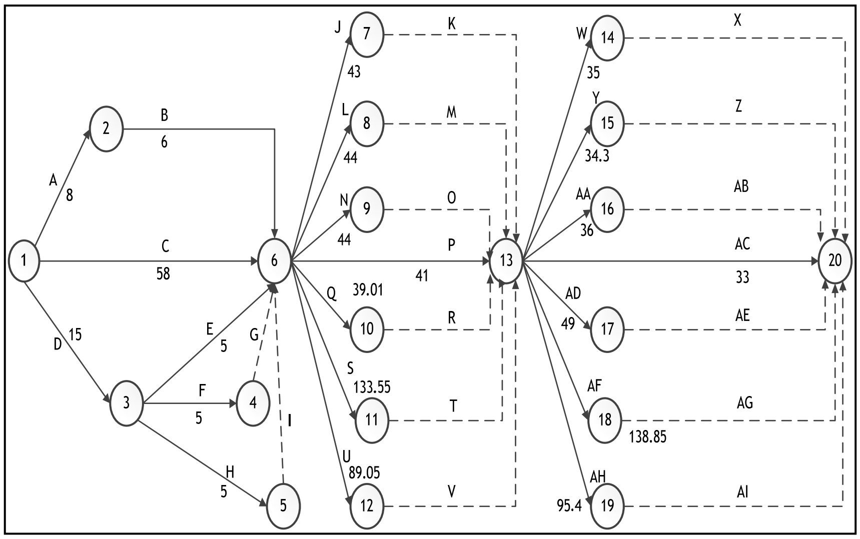

Figure 2: Cost-dependent shortest path problem network diagram

iii. Objective Function

By letting *z* to be the total time for a travel, eqn. 1 to 12 minimize time for each travel as follows:

$$\min z = \sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij}x_{ij} + \sum_{i=m+1}^{n}\sum_{j=m+1}^{n} d_{ij}x_{ij} + \sum_{i=m+n+1}^{p}\sum_{j=m+n+1}^{p} d_{ij}x_{ij} \qquad (1)$$

where,

m, n, p	=	total number of nodes at each stage respectively
i	=	start point
j	=	end point
d_{ij}	=	time travel from city *i* to city *j*

subject to:

For nodes 1 to 6,

$$x_{ij} = \begin{cases} 1, & \text{if } x_{12}, x_{13}, \text{or } x_{16} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \qquad (2)$$

For nodes 2 to 6,

$$x_{ij} = \begin{cases} 1, & \text{if } x_{26} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \qquad (3)$$

For nodes 3 to 6,

$$x_{ij} = \begin{cases} 1, & \text{if } x_{34}, x_{35}, \text{or } x_{36} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \qquad (4)$$

For nodes 6 to 13,

$$x_{ij} = \begin{cases} 1, & \text{if } x_{67}, x_{68}, x_{69}, x_{610}, x_{611}, x_{612}, \text{or } x_{613} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \qquad (5)$$

where, $x_{713}, x_{813}, x_{913}, x_{1013}, x_{1113}, \text{or } x_{1213}$ are dummies

For nodes 13 to 20,

$$x_{ij} = \begin{cases} 1, & \text{if } x_{1314}, x_{1315}, x_{1316}, x_{1317}, x_{1318}, x_{1319}, \text{or } x_{1320} \text{ is selected} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where, x_{1420} , x_{1520} , x_{1620} , x_{1720} , x_{1820} , or x_{1920} are dummies

To calculate the shortest fare for this travel, the objective function (1) is adjusted with y as the minimum total time for a travel and f_{ij} *as* time travel from city *i* to city *j*.

Results and Discussions

Separate calculations were made for TDSPP and CDSPP. As shown in Figure 3, the shortest path for the CDSPP model was marked with red line by starting with node 1, UP. Similar procedures were carried out for TDSPP.

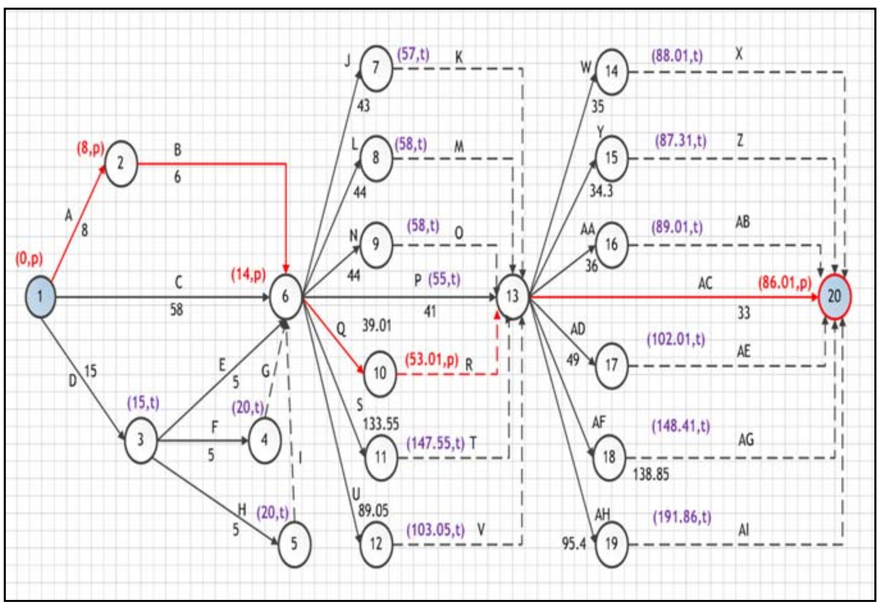

Figure 3: Result in step-by-step Dijkstra algorithm for CDSPP network model

In this research, the shortest path with a total cost of RM86.01 was described by UP – Arau by taxi, Arau – Alor Setar by taxi, Alor Setar – KL by AirAsia, and KL – JB by train. In contrast, the shortest path for TDSPP was identified by the path from UP – Arau by taxi, Arau – Alor Setar by train, Alor Setar – KL by Malaysia Airlines, and KL – JB by Malaysia Airlines. Total time calculated was 160 minutes or 2 hours and 40 minutes.

Conclusion

Due to time constraints on the duration of this study, TDSPP and CDSPP through a fixed sequence of nodes were treated separately. Using step-by-step Dijkstra algorithm, the minimum cost of RM86.01 was achieved in a longer time whereas the cost associated with TDSPP was RM286.40. These findings suggest that travellers have the option of choosing either a time-effective shortest path or cost-effective shortest path. The results from the step-by-step Dijkstra algorithm proved to be similar to results run in C programming. With additional transportations other than the three used in this research, the results would be more interesting. In addition, the scope of this research may be widened to include areas other than transportation problems. Furthermore, the findings would also be interesting if the research were converted to a dynamic SPP by adding float time, or finding one shortest path for both time and cost by adding both SPP.

Acknowledgements
The authors would like to express our gratitude to Transnasional, Sri Maju, City Express, AirAsia, Malaysia Airlines, Malindo Air, and KTMB for allowing access to their data.

References

Abbasi, S., & Ebrahimnejad, S. (2011). Finding the Shortest Path in Dynamic Network using Labeling Algorithm. *International Journal of Business and Social Science, 2*(20).

Jasika, N., Alispahic, N., Elma, A., Ilvana, K., Elma, L., & Nosovic, N. (2012, 21-25 May 2012). *Dijkstra's shortest path algorithm serial and parallel execution performance analysis.* Paper presented at the MIPRO, 2012 Proceedings of the 35th International Convention.

Kamiński, M., Medvedev, P., & Milanič, M. (2011). Shortest paths between shortest paths. *Theoretical Computer Science, 412*(39), 5205-5210. doi: http://dx.doi.org/10.1016/j.tcs.2011.05.021

Lloyd, A. L., & Valeika, S. (2012). Network models in epidemiology: an overview *Complex Population Dynamics* (pp. 189-214): WORLD SCIENTIFIC.

Marasovic, J., & Marasovic, T. (2006, Sept. 29 2006-Oct. 1 2006). *CPM/PERT Project Planning Methods as E-Learning Optional Support.* Paper presented at the Software in Telecommunications and Computer Networks, 2006. SoftCOM 2006. International Conference on.

Murota, K., & Shioura, A. (2014). Dijkstra's algorithm and L-concave function maximization. *Mathematical Programming, 145*(1-2), 163-177. doi: http://dx.doi.org/10.1007/s10107-013-0643-2

Nagurney, A. (2007). Mathematical Models of Transportation and Networks.

Taha, H. A. (2006). *Operations Research: An Introduction (8th Edition)*: Prentice-Hall, Inc.