DECREMENTAL ALGORITHM WITH NODE COMBINATION FOR ROUTING OPTIMIZATION IN ATIS

Pushpi Rani* and Dilip Kumar Shaw**

* **Department of Computer Applications, National Institute of Technology Jamshedpur, INDIA
Email: * pushpi.05.wit@gmail.com, ** dkshaw.ca@nitjsr.ac.in

ABSTRACT: This article discusses the shortest path problem in transportation systems and presents a hybrid approach for routing optimization in ATIS (Advanced Traveller Information System). Due to advancement of technology, routing optimization encounter new challenges day by day. Therefore, a new variant of routing, which combines the decremental approach with node combination is developed to optimize the routing in road network. Proposed method has incorporated into ATIS, aims to provide travellers an optimal path to make an appropriate and better decision. Further the impact of method is evaluated in different aspect such as time and memory and it has been found that the new approach is more comprehensive and memory sparing than the existing one.

KEYWORDS: Advanced Traveller information Systems, Intelligent Vehicle Navigation System, Shortest Path, Node Combination, Decremental algorithm.

INTRODUCTION

The routing problem is one of the most challenging network optimization problems with wide area of applications such as computer networks, communication networks, transportation systems [T. Akiba, Y. Iwata, K. Kawarabayashi, Y. Kawata, 2014]. Furthermore, the routing in intelligent transportation systems (ITS) has gained popularity in last decades**.** ATIS (Advanced Traveller Information Systems), a sub-system of intelligent transportation systems, underlying in this area aims to provide traveller routing information to make a better decision [Mostafa K. Ardakani, 2016]. Nowadays, ATIS is popularly applied which can provide relevant information to travellers so that it is generally believed to be optimal route that is efficient and enhance the performance on traffic networks [Mostafa K. Ardakani, Madjid Tavana, 2015]. ATIS deals with dynamic transportation network and view the road network as the shortest path problem based on travel times change dynamically.

Dijkstra's algorithm is popular and commonly used single source shortest path routing algorithm due to its efficiency and various applications [G. Gallo, S. Pallottino, 1988; D.Z. Du, P.M. Pardalos, 1993; P.M. Pardalos, M.G.C. Resende 2002;]. It traverses the complete graph to find the shortest path from origin node to each node that makes it more complex when implementing the labelling method [P. Festa,2009; T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, 2009]. In this paper, we introduce a hybrid approach, with which the node combination algorithm is applied along with decremental algorithm with certain variations. The Decremental algorithm based on the principle that the traveller always move in forward direction to reach their destination [M.K. Ardakani, L. Sun, 2012], deals with time-dependent cost function with frequent and instantaneous changes; whereas node combination algorithm finds the shortest path with three iterative steps only [Xin Lu , Martin Camitz, 2011]. With Decremental approach, the process of finding the shortest path is more vivid and time consuming; on the other hand node combination method ensure the approach is more comprehensive and memory sparing compare to Dijkstra's algorithm. The approach proposed in this manuscript can be implemented in an Advanced Traveller Information Systems employing the updated traffic information to determine the optimal travel time at each node. It is clear that the real-life traffic never goes according to a priori and static scheme [Mostafa K. Ardakani, 2016].

The paper is organized as follows. In section 2, formal description of Decremental algorithm and node combination algorithm is presented. Section 3, introduces the hybrid approach. Section 4 describes the complexity and evaluates the performance of proposed algorithm with respect to time and memory. Finally the last section summarizes and concludes the paper.

ROUTING MODELLING

A road network can be viewed as a graph where edges represents road segment and nodes corresponds to intersection. A road network is represented by a graph $G = (V, E, P)$, where $V = \{v_1, v_2, \ldots v_n\}$ is the set of nodes, $E = \{e_{ij} | \text{if there is a path from node } v_i \text{ to } v_j\}$ is the set of edges and link characterization P specifying the mean link travel time which are time dependent variables [Mostafa K. Ardakani, Madjid Tavana, 2015]. $\mu_{jk}(t)$ is the mean travel time from j to k at time t and assumed to be available at any time, used to update the link of whole network [M.K. Ardakani, L. Sun, 2012] . Here the assumption is an intelligent vehicle navigation system estimates and provides the expected travel time for any node at any time and the network can be directed or undirected depending on the topology considered for the road network [Mostafa K. Ardakani, Madjid Tavana, 2015]. Figure 1 illustrates a hypothetical network with 11 nodes, i.e., |V|=11, and 17 links.

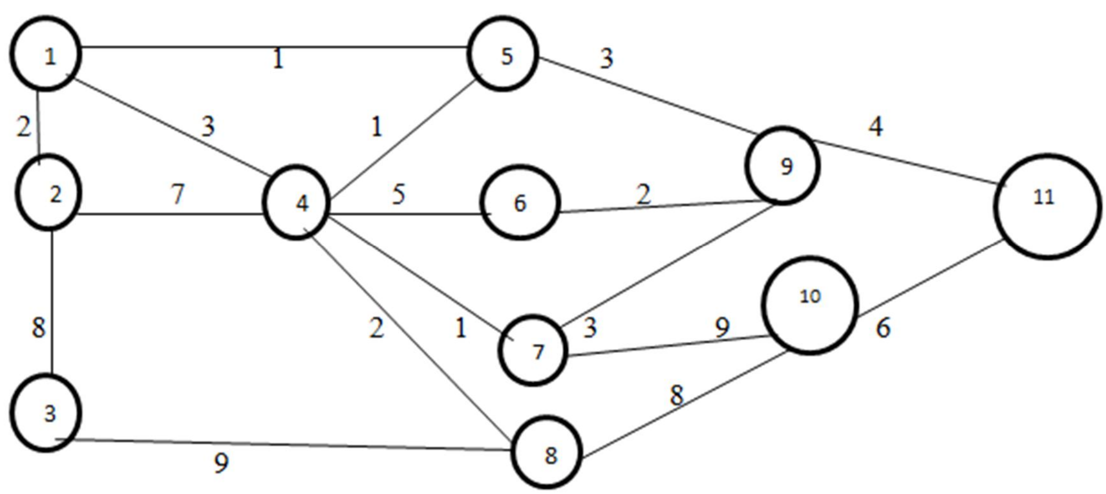

Figure 1: A hypothetical network with 11 nodes and 17 links

Decremental Algorithm

The Decremental algorithm based on the idea that the driver/traveller moves towards the destination, the network size becomes smaller. The fundamental idea behind this algorithm is decrement the network by deleting the noncritical nodes. Noncritical refers to those nodes which are unlikely to be part of optimal route. Here the straightforward approach is label the nodes based on Cartesian coordinate system; the x- (or y-) axis. The origin or source node is labelled as 1 and the other nodes are labelled according to the coordinate system. After node labelling, delete all the links whose label are smaller than the origin node. It is assumed that the chance of finding the better route through backward nodes is small. In this way size of network becomes smaller, and then finds the shortest path from the source node to destination node.
The Decremental algorithm consists of the following steps.

1. *Node labelling* is based on the Cartesian coordinate system; the x-(or y-) axis. The origin label is 1 and the destination label is n.
2. *Delete (j)* delete links having labels smaller than the current node j.
3. *Update (j, k, t)* return updated cost from current node j to node k at time t; for all k belongs to decremented network.
4. *Path (j)* finds the shortest path using Djkstra’s algorithm from current node j to the destination node.

Node Combination Algorithm

Node combination

Definition: Let v_i and v_j are two connected nodes of graph G, the combination of these nodes is the replacement of v_i and v_j with a new node whose incident edges are the edges incident to v_i or v_j. After combination, the number of edges incident to the start nodes will be the sum of edges incident to the combined nodes, less one. Figure 2 show that, after combination new node creates connections with node 5 and 7, which were connected with node 3 before combination.

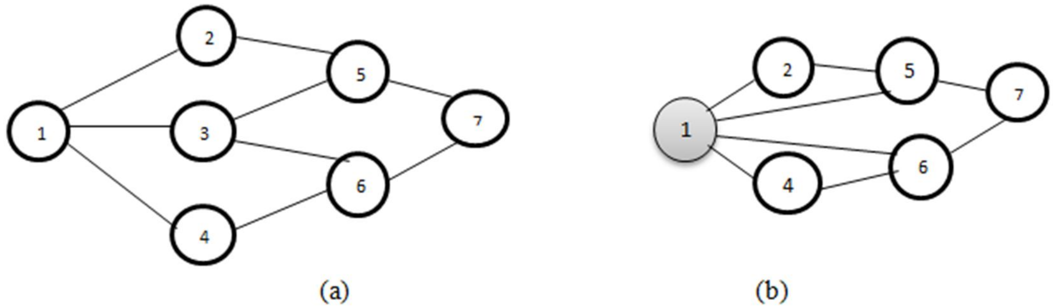

Figure 2: Combine nodes 1 and 3. (a) before combination, (b) after combination

The key idea of NC algorithm is to combine the nodes rather than maintaining the labelling sets in Dijkstra's algorithm. Suppose that all the nodes are connected by a string in a network. The source node is kept in a pool, and it successively dragged other nodes into the pool one by one. The algorithm will terminate when all the nodes have been dragged into the pool.

The combined nodes correspond to the set of nodes for which the distances have been calculated, the adjacent neighbors of the combined node correspond to set of those nodes from which the nearest one is picked. Instead of maintaining vector of distance, distance from the source node is stored by updating the edge weight. This approach makes the algorithm more comprehensive and memory efficient.

The Node Combination algorithm consists of the following steps.

1. *Initialization*. Set $d(V_s) = 0$.
2. *Find the nearest neighbour*. Select V_k from neighbour of V_s. If there is no adjacent node to V_s, stop.
3. *Combine node*. Delete V_k , $V = V - V_k$. If $V = \emptyset$, stop.
4. *Modify egde weights*. For each edge e_{kj} Update W_{sj}. Go to step 2.

PROPOSED APPROACH

The combined approach is simply decrease the size of network by deleting noncritical nodes; update all the links using mean travel time and apply the node combination algorithm to find the shortest path from source to destination. For this algorithm the road network is considered as undirected graph, as it is assumed that estimated mean link travel time required from node A to B can be equal to the time required from B to A.

The recursive equation is formulated to solve this problem is given by

$$g(j) = \min\{\left(g(s) + \mu_{sj}(t)\right), (g(s) + \mu_{su}(t) + \mu_{uj}(t)) \qquad (1)$$

Where, $g(s)$the estimated travel time from node S to node D, $\mu_{sj}(t)$ is the mean link travel time from source node to node j. $\mu_{su}(t)$ *and* $\mu_{uj}(t)$ is the mean travel time from source node to node u and node u to node j respectively. Here u is the node, which is to be combined with source node.

Following are the sequence of steps, for the proposed algorithm.

Step 1. Node labelling is based on the Cartesian coordinate system; the x-(or y-) axis. The label of source node is 0 and the destination label is n.

Step 2. Delete (j) delete links having labels smaller than the source node S.

Step 3. Update (j, k, t) return updated cost from source node S to node k at time t; for all k belongs to decremented network.

Step 4. Compute the Optimal travel time using procedure NC (G_r, S, D).

In step 4 the decremented algorithm is optimized to find the shortest path problem. Here Node Combination algorithm is applied to optimize the problem. Suppose the network is stored as an adjacency matrix, the pseudocode for node combination using status vector V, which identify the nodes that have been combined, can be written as Algorithm 1:

Pseudo code for node combination

Algorithm 1: NC (G_r, S, D)

//***Input***: Decremented graph G_r, source node S, Destination node D.

//***Output***: Optimal travel time from S to D.

$V = V - \{S\}$, u = nearest neighbour of S in V /*Initialization*/

while $(u! = D)$

$V = V - \{u\}$ /*Node Combination*/

for each j in V

$g(j) = \min\{\left(g(s) + \mu_{sj}(t)\right), (g(s) + \mu_{su}(t) + \mu_{uj}(t))$ /*Updating link*/

end for loop

u = nearest neighbour of S in V /*Finding the nearest neighbors*/

end while loop

COMPLEXITY ANALYSIS

From the above pseudo code, the time complexity of NC algorithm is given by $O(E_r + V_r log V_r)$ where, E_r and V_r are the numbers of nodes and links in the decremented network. In decremental algorithm, node labelling operation, is one time operation having complexity $O(V)$. Constant operation is required to delete the noncritical link, its complexity would be $O(c)$, and E_r operation is required to update the link. The total time complexity of decremental algorithm will be $O(E_r)$, by neglecting step 1as it is one time operation and can be considered in pre-processing step. Hence the total running time of the proposed algorithm is given by$(E_r + V_r log V_r)$.
In aspect with space, it is observed that the distances are stored as updated edge weights; therefore, no additional memory is required. In NC algorithm, nodes are combined into the new source node, provided that there is no need to maintain this set (set of nodes). Furthermore, the relaxation is done on the link cost directly, which means that no additional memory is required to record the temporary distances.

CONCLUSION

This paper addresses a real life transportation problem that will become increasingly important results more intelligent transportation systems are employed. Therefore, the hybrid routing approach with certain assumptions is proposed for a network based on link travel time. The main focus of this paper is memory efficiency along with run time efficiency. The node combination algorithm is embedded with decremental approach to speed-up the optimization process, as the decremental approach is not guaranteed to find the optimal path due to heuristic in nature. Although the effect of decremental algorithm delete the noncritical nodes results to reduced network size. NC algorithm is an alternative way of Dijkstra's algorithm, which is more straightforward, simple and memory-sparing. The Node Combination along with the decremental approach can be applied in an Advanced Traveller Information system to provide drivers with minimum travel times, better path, and more realistic estimates.

REFERENCES

[1] Du D.Z., Pardalos P.M. (Eds.); (1993) *Network Optimization Problems: Algorithms, Applications and Complexity*; World Scientific, Series on Applied Mathematics;2

[2] Gallo G, Pallottino S.; (1988); *Shortest path algorithms*; Annals of Operations Research; 13 (1) ;1–79

[3] Ardakani M. K; *Adaptive MDS-based algorithm for dynamic routing optimization in advanced traveler information systems*; (2016); World Journal of Modelling and Simulation; 12 (1); 15-23

[4] Ardakani M. K., Sun L; (2012) ;*Decremental algorithm for adaptive routing incorporating advanced traveler information;* Computer &. Operation Research; 39 (12); 3012–3020

[5] Ardakani M. K , Tavana M; *A decremental approach with the A* algorithm for speeding-up the optimization process in dynamic shortest path problems*;(2015); Measurement; 60; 299–307

[6] Festa P.; (2009) *Shortest path tree algorithms*; Encyclopedia of Optimization, Springer (2); 3507–3519.

[7] Pardalos P.M., Resende M.G.C. (Eds.); (2002) ;*Handbook of Applied Optimization*; Oxford University Press

[8] Akiba T., Iwata Y., Kawarabayashi K., Kawata Y; (2014);*Fast shortest-path distance queries on road networks by pruned highway labelling ;* Proceedings of the 16th Meeting on Algorithm Engineering and Experiments (ALENEX'14), SIAM, 147–154.

[9] Cormen T.H., Leiserson C.E., Rivest R.L., Stein C.;(2009); *Introduction to Algorithms*; third ed., MIT Press

[10] Lu X, Camitz M; (2011); *Finding the shortest paths by node combination*; Applied Mathematics and Computation; 217 6401–6408