GIS-Based Itinerary Planning System for Multimodal and Fixed-Route Transit Network

QIANG LI AND CARL E. KURT

This paper introduces a multi-objective linear programming model for transit itinerary planning (TIP) with multimodal and fixed-route transit networks and presents an efficient two-phase TIP algorithm to find the optimal path that has the least combination cost from a given origin to a given destination. The algorithm recognizes the inherent nature of the multi-objective and time schedule constraint of TIP and considers trade-off among multiple optimization criteria in the path selection process. In particular, the algorithm of K shortest path problem with multiple time windows associated with time schedules is proposed in order to generate a set of path alternatives for evaluation and choice of the best path. A GIS-based Transit Itinerary Planning Decision Support System (GIS-TIPDSS) for assist passengers with itinerary decision making was developed. The GIS-TIPDSS was tested using data from a real transit network.

INTRODUCTION

Pre-trip transit information systems (PTIS) are the important component of advanced traveler information system (ATIS) and a means of alleviating the uncertainty about transit schedules and routes (*1*). Pre-trip information can cover a wide range of categories, including transit routes, maps, schedules, fares, and more. PTIS utilizes this information to support itinerary planning and helps travelers to make decisions on the itinerary path from given origin to destination, even if the trip involves multiple modes. The itinerary decision may involve transfers among different modes or routes and scheduling at the transfer points. This demands significant effort and time in a complicated multimodal transit network. PTIS allows travelers direct access to transit route information and improves travelers' travel time by providing the optimal path between the origin and destination. PTIS may also attract more potential passengers to use the transit system because of its user-friendly interface and efficient routing.

In general, transit itinerary planning (TIP) is a multi-objective decision making process. The solution is rather complicated because of multiple decisions, multiple criteria, and uncertainty. The decisions depend on many factors such as in-vehicle travel time, walk time, transfer time, number of transfers, reliable schedules, etc. Because of the multi-objective nature of TIP and conflict criteria in problem solving, there may be no single optimal solution, but rather a group of potential best solutions, from which the decision maker selects the best compromise in an heuristic fashion.

Q. Li, Transportation Center of Kansas University, Lawrence, Kansas 66045. C. Kurt, Civil and Environmental Engineering, University of Kansas, 2103 Learned Hall, Lawrence, Kansas 66045.

In this paper, we propose a multi-objective linear programming model for TIP and a corresponding two-phase heuristic solution algorithm, which is built on a transit network representing a multimodal and fixed-route transit system. The two-phase algorithm is a heuristic solution of the multi-objective linear programming model. The first phase generates feasible path alternatives between given origin and destination points, with the aim of minimizing the total travel time including both travel and waiting time. The second phase is to evaluate these feasible paths and select the best path. The evaluation of the alternatives is based on a linear disutility function, which takes into account decision criteria such as number of transfer points, bus headway or frequency, total travel cost, etc.

One of the purposes of the research is to integrate the developed TIP model and geographic information system (GIS) technology. The GIS-TIPDSS was designed and implemented within a GIS environment, MapInfo for Windows. GIS plays a key role in integrating transportation data and analysis models. This paper describes the development of the GIS application system and related with components.

PROBLEM DEFINITION AND FORMULATION

Conceptually, TIP finds an optimal path from a given origin to a destination with specific departure or arrival times subject to certain constraints in a fixed-route transit network. The objectives of the routing problem may concern minimization of total travel time, number of transfers, number of modal changes, and trip cost. The time constraint considers that a vehicle arrives at a prescheduled transit station—called a time point—with a list of departure times and requires departure from the station at the next departure time. Therefore, the time constraint can be treated as multiple time windows bounded by scheduled departure time lists.

Modeling Multimodal Transit Network

A multimodal transit network may include bus, metro, and subway, where service routes are fixed and the departure or arrival at certain stations is scheduled in advance and generally not subject to changes. A derived network model must capture all possible fixed-route transit modalities and the interconnections among them. In particular, the network model must represent accessibility between any two modes or any two routes. The derived transit network consists of a set of nodes representing prescheduled stations, or time points, a set of transit route segments connecting two nodes on the same route, and a set of transfer links connecting two nodes from different routes.

In addition to representing physical elements (e.g. route segments and transit stations) of the transit network, the network model also must consider the characteristics and activities of the fixed-route trips. Different virtual links, called transfer links, are added to the network graph to represent the following activities: access from traveler's origin to a feasible transit station, waiting for a bus or a train, boarding/alighting a bus or a train, walking between two transit stations for transfer, etc.

Let G = (N, L) be a multimodal transit network, where N is the set of *n* nodes and L is the set of *m* direct links (i, j), connecting node i to node j in the network. Each distinct node represents specific route station and is assigned a route ID. The network model includes two types of links, specifically:

1. Route links, which correspond to segments of transit routes between consecutive prescheduled transit stations on the routes.
2. Transfer links, which represent transfer time from one station to another, plus alighting/boarding time at the two stations.

Each link (i, j)ÎL has a weight associated with it given by the time (cost) required to travel or transfer from node i to jÎN. In particular, let c_{ij} be the weight representing the sum of time (cost) traveling from node i to node j if the link (i, j) is a route link, or the sum of the time spent on transferring and dwelling if the link (i, j) is a transfer link.

Basically, we consider three attributes related to the travel cost, namely, in-vehicle riding time, average stopping time representing estimated stopping time at unscheduled transit stops or unexpected stops, and riding fare converted into time units. Similarly, by definition of transfer link, the transfer cost includes walking time from one station to another station and the boarding/alighting time. The representation of the transit network with weights allows one to search optimal paths, taking various costs into account.

Formulation

As mentioned before, the TIP, specifically, the two-point optimal routing problem with multiple time windows, can be formulated as a multi-objective integer programming problem given a set of optimal criteria and constraints. Basically, the following optimality criteria are considered in the mathematical programming model:

- minimizing total travel time, including riding and dwell time;
- minimizing total waiting time occurring at prescheduled nodes;
- minimizing path disutility cost caused by transferring activities.

Unfortunately, it is very difficult and expensive using current solution methods to deal directly with the multi-objective integer programming model. There are two major reasons for this. First, the problem formulation cannot be solved directly in a reasonable amount of time because the routing problem is an NP-hard problem, and there are no known polynomial time algorithms for the exact solution of the problem. Second, the optimality criteria include multiple objectives with complicated and even conflicting constraints. There may be no absolutely optimal solution; rather a best-compromised solution may be preferred. Therefore, a heuristic approach was taken instead of directly solving the mathematical programming models.

HEURISTIC SOLUTION AND ALGORITHM

Due to the considerable complexity of the TIP model, a heuristic solution is regarded as a natural approach for solving the problem. To develop the heuristic solution for the TIP, the TIP is considered as consisting of the following two major related subproblems:

- A K-shortest path problem (KSPP) for generating a set of route alternatives between given origin and destination.
- A route choice problem, where the generated path alternatives are evaluated based on a route disutility function and the best compromise is selected.

The composite heuristics were developed to obtain the best-compromised solution by combining route's alternative generation with route selection. The first step identifies K shortest paths on the network between given origin and destination using a modified label setting technique. Then, the best-compromised path is selected in the set of the generated path alternatives based on the optimal criteria.

Modified K Shortest Path Algorithm

Using the mathematics notation in previous section, let o and d be given origin node and destination node, a path from node o to node d is an ordered sequence of nodes and links, namely, $\{n_1 = o, (n_1, n_2), n_2, (n_2, n_3), n_3, \ldots, (n_{k-1}, n_k), n_k = d\}$, where ni $\in$ N, $(n_{i-1}, n_i) \in L$ for any i = 1, 2, …, k-1. Let P_{od} denote the set of paths from node o to node d in G (N, L) and $T(P_{od}) = \{t(p_1), t(p_2),\ldots, t(p_k)\}$ denote the set of corresponding path times (costs). The objective of a single shortest path problem is to determine a path $p_{od}^* \in P_{od}$, for which $t(p_{od}^*) \le T(P_{od})$ holds for any path $p_{od} \in P_{od}$. If one requires not only to find the dominant shortest path but also to determine the second, third, …, u_p to K[th] shortest path between a given origin and destination, the shortest path model extends to calculate K shortest paths listed by non-decreasing order of the objective values. So, the KSPP is to determine a set of $P_{od}^k = \{p_1, p_2, \ldots, p_k\} \in P_{od}$ such that:

$$t(p_i) \le t(p_{i+1}),\ for\ any\ \ p_i \in P_{od}^k \qquad (1)$$

$$t(p_k) \le t(p\ \ ),\ for\ any\ \ p_i \in P_{od} - P_{od}^k \qquad (2)$$

To tackle the KSPP with time constraint, a modified KSPP algorithm with multiple time windows was developed. The model identifies a set of feasible K shortest paths, which considers riding time and stopping time along a link, and waiting time at node.

Basically, this KSPP model is a modified version of label setting KSPP algorithm (*2*). Shier's (*3*) study indicated that the label-setting algorithm required the least computation time for low density or sparse networks such as transportation networks. The modification of the KSPP model is related to addition of multiple time windows as time constraints. A vehicle that arrives at a prescheduled node requires departure only at one of scheduled departure time list. As a result, a waiting time may occur.

The modified label-setting algorithm solving the KSPP iteratively is presented in detail below. Assume for a given directed transit network G(N, L), a timetable storing schedule list, a pair of origin o and destination d, and a travel time matrix $T = \{t_{ij}\}$ are available. For each node i, two k-vector labels, arrival time and departure time, denoted by

$A_i = \{a_i^1, a_i^2, \ldots, a_i^k\}$ and $D_i = \{d_i^1, d_i^2, \ldots, d_i^k\}$, $i \in N$, are attached. The entries of A_i and D_i are listed in increasing or at least non-decreasing order, respectively. The two k-vectors represent the earliest arrival and departure time, second earliest arrival and departure time, up to the k^{th} fastest path starting from node o to node i. Each entry of the two k-vectors and corresponding node have two states, permanent or temporary. Initially, all entries are set to temporary. If all entries of k-vector of node i become permanent, the k-shortest paths from node o to node i are found and node i is set to permanent. In order to speed up the calculation and manage the labels efficiently, the minimum labels of D_i are maintained in a priority queue, specifically, a binary heap, $Q = \{q_1, q_2, \ldots, q_n\}$, where q_i represents the minimum value of temporary labels associated with a node departure label vector. Once all entries of D_d vector become permanent, the search of k-shortest paths from the origin to destination is complete. The entry values of A_d indicate the k shortest travel times in increasing order.

Algorithm Steps

Step 1: Initialize network. For all node $i \neq o \in N$, set $A_i = \{\infty, \infty, \ldots, \infty\}$ and $D_i = \{\infty, \infty, \ldots, \infty\}$, $A_o = \{s_t, \infty, \infty, \ldots, \infty\}$ and $D_o = \{s_t, \infty, \infty, \ldots, \infty\}$, where s_t is the starting time at node o. Insert s_t into the priority queue ($q_1 = s_t$) and set d_s^1 permanent. Then, set all entries except entries of node o to temporary.

Step 2: Let $l = q_1$ associated with node n*, for each node j adjacent to node n*; 2a: If $l + d_{n^*j} <$ any $A_j = \{a_j^1, a_j^2, \ldots, a_j^k\}$, replace the first encountered entry (a_j^m),which greater than $l + d_{n^*j}$, by ($l + d_{n^*j}$); 2b: If $dt_{t-1}(j) < a_j^m \leq dt_t(j)$, replace the corresponding d_j^m in D_j with $dl_t(j)$. Otherwise, $d_j^m = \infty$, indicate that no scheduled departure time in time list is available when vehicle arrival at node j at time a_j^m; 2c: If updated value of d_j^m is smaller than node j's entry in the priority queue, replace node j's entry by d_j^m and reorder the queue.

Step 3: Set the entry of A_{n^*} corresponding to l permanent. If all entries in A_{n^*} become permanent, remove the node n* from the priority queue. Otherwise, replace node n*'s entry in the priority queue by next remaining temporary label in D_{n^*} and reorder the queue.

Step 4: If the removed n* = d, stop iteration. Otherwise, go to step 2.

At the end of this iteration, the arrival time vector at node d (A_d) gives the minimum total travel time, second minimum total travel time, third, up to k^{th}, required from origin o to node i by going through only the schedule and path feasible nodes.

Paths Evaluation and Selection

In the real world, transit passengers choose the best path by considering not only the usual link-based shortest criteria but also non-link based factors, which could affect their path choice, such as average headway (frequency) along the path, total number of transfer points along the path, and total path cost (e.g. fare). It is difficult and even impossible to implement the path-based attributes into the link-based cost functions for shortest path calculation in label setting shortest path algorithm. This is why the objective function is presented as three separate optimal criteria: link-based travel time, node-based waiting time associated with the specific entry link, and path-based disutility cost. After the K shortest path model takes care the first two optimal criteria, this evaluation algorithm will select a path with minimized disutility cost.

The algorithm for path evaluation and selection is simple. For each generated shortest path, the value of the disutility function associated with the path is calculated. The comparison between the values of the disutility is performed. Finally, the best optimal path is selected in the K generated paths.

DEVELOPMENT OF GIS-TIPDSS SYSTEM

Design of Transit Itinerary Planning Systems requires a computer tool such as GIS that can integrate and maintain large-size spatial transportation databases from different data sources and can conduct and support spatial and temporal analysis. Particularly, GIS has the ability to model and refine large-scale networks and control quality of information flow among various models. To integrate TIP model and GIS technologies, the functionality of a GIS system needs to be extended or modified. The key to the successful integration is the design of spatial network databases and associated management tools to meet the various spatial and temporal functions needs of TIP.

GIS-TIPDSS was implemented on a personal computer and intended for potential transit users. There are three main modules consisting of the GIS-TIPDSS: input module, TIP module, and output module. The input module includes preprocessing passenger information and loading transit network. The TIP module performs the two-phase itinerary planning and produces the best-compromised path. The output module includes post-processing the results from TIP module and displaying the best path. Input and output modules were designed and developed using the MapBasic provided by MapInfo. The routing module was coded using Visual C^{++}.

The input and output modules provide interfaces between the GIS-TIPDSS and users. Since MapInfo provides the GUI development platform and common parameter standards, the user-friendly GUI is incorporated well into the GIS-TIPDSS system. The general principle of the GUI design is that the GUI utilizes the MapBasic program to maximize the use of built-in MapInfo utilities. The users or passengers can enter all the information pertaining to a desired trip such as origin/destination addresses and either arrival or departure times, etc. The input module processes the passenger's input information. The process includes finding feasible and accessible transit stations closest to the given origin and destination and with the earliest departure time and then passing the info on routing module. Based on the time and date required for riding service, the input module will load available transit routes with corresponding time schedule table. The following are the basic steps:

Step 1: Locate the origin and destination on the map using the GIS address geocoding;

Step 2: Identify the nearest transit stations to the origin and destination locations;

Step 3: Check the feasibility of schedule time of the selected stations and determine the proper departure time at starting station.

The TIP module is the key component of the GIS-TIPDSS. It calculates and recommends the best-compromised path based on the requirement of passengers. From the input module, the passenger's spatial and temporal information is passed on the TIP module. An optimal path is produced from the TIP module.

The path planning proceeds as follows:

Step 1: Calculate K shortest paths from the origin node to the destination node, minimizing in-vehicle riding time and waiting time. These paths are stored in ascending order with respect to total travel time along path;

Step 2: Set k = 1 and Set p* = ∞; retrieve the k^{th} shortest path, calculate the disutility function over the path, and then store the function value in p*;

Step 3: Set k = k + 1, If k > K, go to step 4; otherwise, retrieve the k^{th} shortest path, calculate the disutility function over the path, if the value of the function < p*, replace the value of p* by the value of disutility function. Back to start of the step 3;

Step 4. Take p* as the optimal path, and stop.

The result p* from the TIP module is passed to the output module. The passengers get visualized best path between the given origin and destination on transit network map.

The GIS-TIPDSS was tested using a real transit network, Las Vegas Transit Bus System. The network consists of 4336 stops and 549 time points. The average number of transit running routes each day is about 85. The TIP algorithm was enhanced to consider more complicated scenarios because of the case study. The average time needed to find an optimal path solution from a given origin and destination was about 6 seconds. By testing the real network, it was demonstrated that the two-phase TIP algorithm is capable of producing an optimal path solution in a real multimodal transit network within a reasonable time frame.

CONCLUSION

In this paper, the TIP was modeled as a multi-objective routing-selection problem. The heuristic methodologies and algorithm were developed for generating the path alternatives and selecting a best path by evaluating the alternatives. The algorithm is a two-phase procedure, which is designed to simulate the decision making process for TIP. A GIS application for TIP to assist passengers with itinerary decision making was developed.

REFERENCE

1. Kikuchi, Shinya. *Advanced Traveler Aid Systems for Public Transportation*. Report DOT-T-95-07. FTA, U.S. Department of Transportation, 1994.
2. Chen, Y.L. *Finding the K quickest Simple paths in a Network*. Information Processing Letters, Vol. 50, 1994, pp. 89-92.
3. Shier, D. On Algorithms for Finding the K Shortest Paths in a Network. *Networks*, Vol. 9, 1979, pp. 195-214.