A fast method for estimating railway passenger flow

Y. Nagasaki, M. Asuka & K. Komaya
Advanced Technology R&D Center, Mitsubishi Electric, Japan

Abstract

To improve the situation for crowded commuters in Japan, it is important to plan a train schedule that considers passenger behavior, such as their choice of trains and the transfer stations used to reach their destinations. However, it is difficult to directly measure such detailed behavior using the present infrastructures, with which we can only get OD (Origin-Destination) data from the automatic ticket gates. The obtained OD data only consists of the number of passengers for each origin-destination and the time each passenger passes through the gates. In this article, to contribute to the planning phase of a new train schedule, the authors propose a method for estimating railway passenger flow using OD data. This paper firstly points out that the problems of estimating passenger flow can be boiled down to a shortest path problem of graph theory by assuming a certain passenger behavior model. By representing train operations in a graph structure, we can assume that a passenger will use the minimum cost path to his/her destination. This paper secondly proposes a method for conducting fast searches of the graph structure. The method uses the fact that railways operate on a time schedule. This method can estimate passenger flow fast enough so as to apply it to a practical train schedule planning support system. Lastly, the authors show the results of applying the passenger flow estimation system to a railway in an urban area in Japan.
Keywords: train schedule evaluation, passenger flow estimation, graph theory, shortest path problem, incremental assignment procedure.

1 Introduction

Currently, planning train schedules depends mainly on expert planners' experiences and intuitions, although a computer based decision support system is gradually appearing. To promote the use of a computer assistance system in a

WIT Transactions on The Built Environment, Vol 88, © 2006 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/CR060181

train schedule planning phase, it is important to evaluate train schedules quantitatively. It is easy for a computerized system to evaluate train schedules from the viewpoint of a railway operator, for example the total travel distance of rolling stocks. However, when experts are planning passenger train schedules, they must consider not only train movements but also passenger behaviors. They must also leave enough transfer time, consider each train's congestion rate, and so on. However, these considerations are very qualitative and do not necessarily suit computerized systems. In order to construct a better decision support system, train schedules must be quantitatively evaluated from the viewpoint of passengers.

To complete such an evaluation, we need detailed information about passenger behaviors such as train choices and the transfer stations used to reach a particular destination. Passenger behaviors depend on a train schedule. We can not obtain such information about passenger behaviors by measuring until trains are operated on the schedule actually. Thus, to evaluate a train schedule in the planning phase, a passenger flow estimation method is required.

In the field of passenger flow estimations, the following methods are currently used.

- A method employing a graph theory's shortest path algorithm [1].
- A method employing network user equilibrium model [2].
- A method employing a logit model [3].

In this paper we propose a fast method for estimating railway passenger flow from a train schedule and OD (Origin-Destination) data. OD data consists of the number of passengers for each origin-destination and the time each passenger passes through the automatic ticket gates. This method is based on a graph theory's shortest path algorithm and it has been improved to work faster than conventional algorithms by using the fact that a railway operates on a time table.

2 Passenger flow estimation using a shortest path algorithm

2.1 The passenger behavior model

From the departing station to the destination station, a passenger can often select his/her route from multiple possibilities. A passenger generally prefers a route that takes the least time, with the least transfer barriers. In this research, we assume that a passenger will select the route that has the least evaluation value. The evaluation value for a route is the sum of the required time of the route and the transfer barrier conversion values. The transfer barrier conversion values represent passenger inconvenience in the dimension of time. That is, we assume that a passenger weighs the inconvenience of having to transfer against the amount of time saved by making a transfer.

2.2 A graph structure representing passenger routes

A graph structure representing passenger routes is created based on a train schedule. The vertex in the graph structure represents each train's departure or arrival at a station. The directed edge from the departure vertex to the arrival

vertex represents the movement of a train between stations. The directed edge from the arrival vertex to the departure vertex represents a train stopping at a station or passengers transferring between trains at a station. Each edge has a cost value that quantifies the degree of passenger inconvenience when choosing the edge. The cost value for a movement edge or a stop edge is the time required of the act represented by the edge. The cost value for the transfer edge is the sum of the time required for the transfer and the conversion value of the transfer barrier. An example of such a graph structure is shown in Figure 1. In such a graph structure, based on the passenger behavior model, a passenger will use the route with the minimum cost from the departure vertex of his/her departing station to the arrival vertex of his/her destination station. Therefore, the shortest path algorithms of the graph theory can be applied to find each passenger's route.

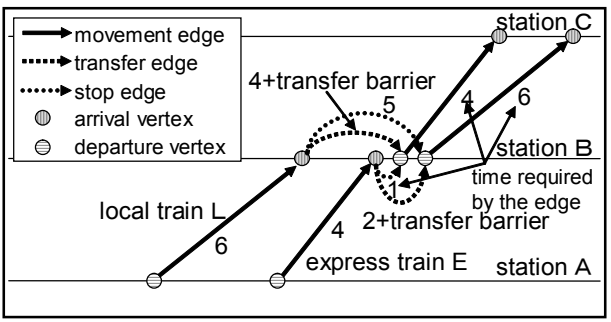

Figure 1: A graph representation of passenger routes.

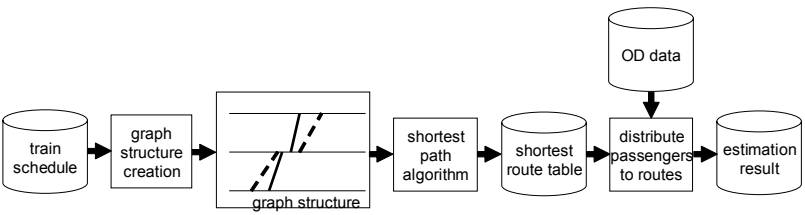

Figure 2: Architecture of passenger flow estimation.

2.3 Applying the shortest path algorithm and estimating passenger flow

Dijkstra's algorithm is often used to find the shortest path from a designated vertex to the other vertices in a graph structure. Previous works simply applied Dijkstra's algorithm on the graph structure that represented passenger routes. We can obtain passenger route tables for any combination of departure and arrival

station by applying Dijkstra's algorithm from each departure vertex at each station.

When a passenger's departing station, destination station, and arrival time at the departing station are designated, we can estimate his/her route to the destination station by looking up the route table. We can also estimate whole passenger flow by accumulating each passenger's route.

The total flow of passenger flow estimation is shown in Figure 2. Using this estimation architecture, the congestion rate of each train in each section can be estimated from the train schedule and OD data.

3 Optimization of the shortest path algorithm using the features of the railway operation

3.1 Conventional method problems

The conventional method repeatedly applies Dijkstra's algorithm on the graph structure. In the process, the algorithm repeatedly executes almost the same calculation on the same part of the graph structure. To decrease the calculation requirements of the whole process, such useless iterations should be reduced.

The conventional method only uses the features of general graph theory and doesn't use the features of the railway operation. We propose a system that uses the optimized shortest path algorithm and that includes the features of railway operation.

3.2 Total flow of passenger flow estimation using the optimized shortest path algorithm

Figure 3 shows the total flow of passenger flow estimation using the optimized shortest path algorithm. The difference between it and conventional model is that we have changed the shortest path algorithm from Dijkstra's algorithm and added the preparative search algorithm against the graph structure.

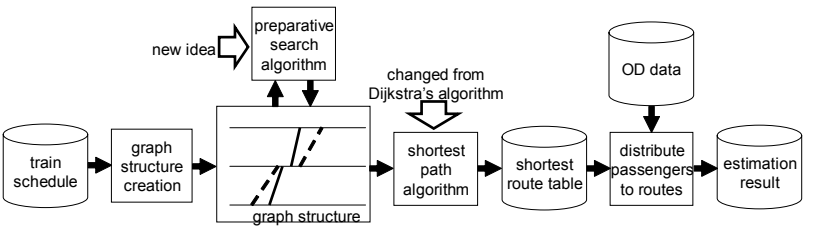

Figure 3: Passenger flow estimation architecture using the optimized shortest path algorithm.

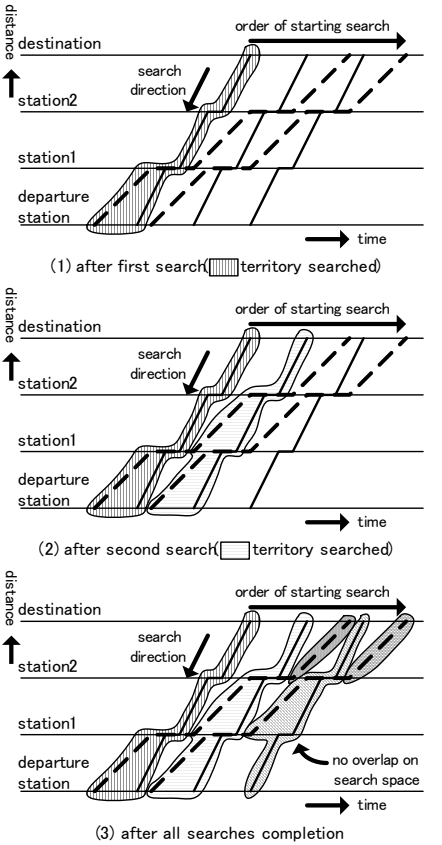

Figure 4: The concept behind the backward preparative search algorithm.

3.3 The backward preparative search algorithm

In the new method, after the graph structure is created, the backward preparative search algorithm is applied to the graph structure. Figure 4 illustrates the concept behind the backward preparative search algorithm. A preparative search starts from the destination station's arrival vertex and tracks back to each directed edge in reverse. From the vertices that are currently searched, a passenger can reach the arrival vertex where the search starts, because a search always tracks back to the directed edges in reverse. The searched vertices are marked to classify the destination station's vertex at the point where the search starts.

Preparative searches are processed in the sequence of destination station's arrival times. A later search should not overwrite nor track back to vertices that are already marked by previous searches. From vertices that are already marked

by previous searches, a passenger can reach an earlier arrival vertex for the destination station, because searches are processed in the sequence of the destination station's arrival time. Therefore, after all preparative searches are completed, each vertex is classified by the earliest arrival vertex of the destination station that a passenger can reach from the vertex.

In a case where the cost of a transfer edge contains the transfer barrier's conversion value, the route by which a passenger can arrive at their destination in the shortest time is sometimes not the most convenient. Thus, the marking on the vertices should be recorded by an evaluation value that shows the best route to a destination. Figure 5 shows vertices with their evaluation values of the best route to a destination. At the beginning of a search, the arrival vertex from the destination station should be marked as 0. When a search tracks back to directed edges in reverse, the evaluation value of the precedent vertex is recalculated by adding the cost of the edge to the evaluation value of the subsequent vertex. If the current evaluation value of the vertex is greater than the value newly calculated, a search can rewrite the evaluation value of the vertex and continuously track back to the precedent edges of the vertex.

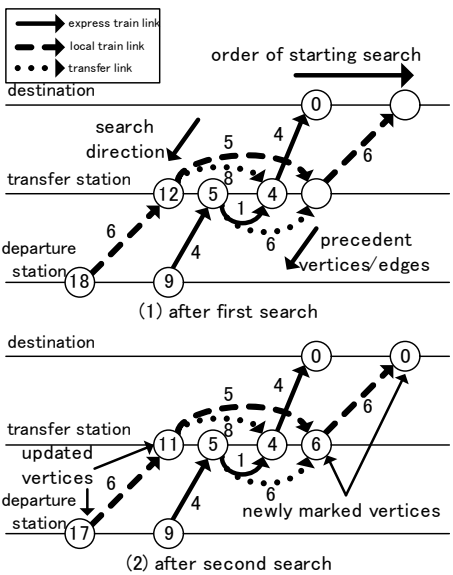

Figure 5: Calculating the evaluation values for vertices using a backward preparative search algorithm.

3.4 The optimized shortest path algorithm

After all backward preparative searches are complete, the optimized shortest path search algorithm is carried out. A search starts from the departure vertex of a departure station and follows the directed edges in a forward direction. During the forward search, the edges that should be searched can be easily identified,

because the evaluation value given to each vertex shows remaining distance to the destination. If the cost value of an edge is not equal to the deference of the evaluation value of both end vertices, the edge doesn't need to be searched. Because the algorithm doesn't follow nonessential edges, it can be processed faster than the simple Dijkstra's algorithm.

The new algorithm proposed in this paper resembles Bellman's Dynamic Programming in that a backward preparative search labels each vertex and later a forward search finds shortest paths. The difference is that the new algorithm uses the features of railway operations to reduce the search space of the preparative backward searches.

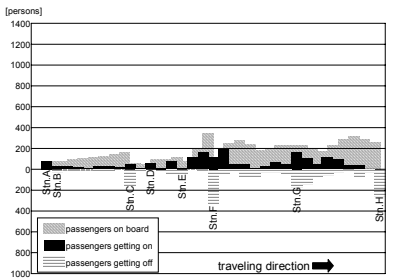

Figure 6: The estimated number of passengers of the local train in the morning.

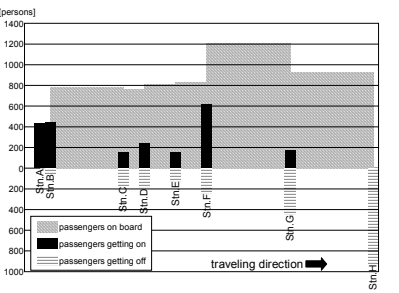

Figure 7: The estimated number of passengers of the express train in the morning.

4 The results of the passenger flow estimation method applied to a railway system in an urban area in Japan

4.1 The estimation parameters

We applied the passenger flow estimation method proposed in this paper to a railway system in an urban area in Japan. The target railway network operates about 1000 passenger trains per day. We used a weekday train schedule and OD data that was taken from automatic ticket gates.

4.2 The estimation results

We executed the estimation on a Pentium 4 3GHz PC; it took about 14 seconds to complete. The conventional method using the simple Dijkstra's algorithm took about 10 minutes against the same data.

Figure 6 and Figure 7 show the estimated number of passengers on the local and express trains in the morning. Stations are shown horizontally and the trains travel from left to right. The upward bar graph from center shows the number of passengers getting on the train at each station. The downward bar graph from

center shows the number of passengers getting off the train at each station. The background bar graph shows the total number of passengers in the train at the time of departure from the station.

5 Congestion-sensitive estimation using the incremental assignment procedure

5.1 The congestion-sensitive estimation method

The passenger behavior model shown in 2.1 assumes that passengers will select their routes according to the journey time and transfer barriers. However some passengers avoid riding on over-crowded trains. Such passenger behavior can be represented by adding an evaluation value for the train congestion to the edge costs. Even so, each train's congestion rate can be calculated only after the passenger flow estimation is complete. To avoid loop references, we use the incremental assignment procedure. The whole passenger demand is divided into several parts. For the first part, the passenger flow estimation is calculated normally. Based on the first result, the costs of edges are changed and passenger flow estimation is calculated for the second part. One by one, the edge costs are changed and the passenger flow estimation is repeated. From this, the total of all these parts gives us the result.

Eqn (1) is a congestion rate to time conversion function taken from [4].

$$f_c'(r) = \max\left(0.807 \cdot \left(r/100\right)^{1.112} - 1,\ 0\right). \tag{1}$$

In eqn (1), r means the congestion rate of a train. When the congestion rate is specified, the function returns a coefficient value. If a returned coefficient value is 2, this means that a passenger on a train for one minute will feel as the same as a passenger on an empty train for two minutes.

Figure 8 shows the estimated number of passengers with and without using the incremental assignment procedure in some trains in a section. The number of divisions in the incremental assignment procedure is eight. The illustrated part is during the morning rush-hour, therefore some express trains are very crowded. Using the incremental assignment procedure, some passengers are estimated to avoid riding crowded trains such as Express 2, and use other trains, such as Local trains, instead.

6 Conclusions

In this paper, we have shown a fast method for estimating railway passenger flow. This method can complete estimations faster than the conventional methods using the features of railway operations. This method can also estimate congestion-sensitive passenger flow using the incremental assignment procedure. We can evaluate a train schedule from the viewpoint of passengers by using this method. We can also make train schedules with better services for passengers by calculating the estimated passengers' flow.

References

[1] Nagasaki, Y., Takano, M., & Koseki, T., Train Rescheduling Evaluation and Assistance System with Passengers' Behavior Simulation Based on Graph Theory, IEEJ Technical Meeting on Systems and Control, SC04-11, pp.25-30, 2004.
[2] Ieda, H., Akamatsu, T., Takagi, J.,& Hatakenaka, H., Commuter's Benefit Evaluation of Train Scheduling by Network User Equilibrium Model, Infrastructure Planning Review No.6, pp.177-184, 1988.
[3] Hirai, C.& Tomii, N., A Method to Estimate the Number of Passengers for Evaluation of Traffic Rescheduling Plans, RTRI Report Vol.14 No.7, pp.43-48, 2000.
[4] Yamaga, H.& Hatta, T., Measurement of Commuter's Prostration Cost and Optimal Congestion Charge, The Japan Center for Economic Research Bulletin No.41, pp.110-131, 2000