

Research Article

Optimal Path Choice in Railway Passenger Travel Network Based on Residual Train Capacity

Fei Dou,¹ Kai Yan,¹ Yakun Huang,¹ Li Wang,² and Limin Jia²

¹ School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

² State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Limin Jia; jjalm@vip.sina.com

Received 17 January 2014; Revised 7 May 2014; Accepted 3 June 2014; Published 6 July 2014

Academic Editor: Dimitrios A. Karras

Copyright © 2014 Fei Dou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Passenger's optimal path choice is one of the prominent research topics in the field of railway passenger transport organization. More and more different train types are available, increasing path choices from departure to destination for travelers are unstoppable. However, travelers cannot avoid being confused when they hope to choose a perfect travel plan based on various travel time and cost constraints before departure. In this study, railway passenger travel network is constructed based on train timetable. Both the generalized cost function we developed and the residual train capacity are considered to be the foundation of path searching procedure. The railway passenger travel network topology is analyzed based on residual train capacity. Considering the total travel time, the total travel cost, and the total number of passengers, we propose an optimal path searching algorithm based on residual train capacity in railway passenger travel network. Finally, the rationale of the railway passenger travel network and the optimal path generation algorithm are verified positively by case study.

1. Introduction

Railway as a kind of large volume passenger transportation modes has been developing in China and other countries (e.g., USA and Europe) in an even larger scale, and it plays a role of the passenger transportation systems. How to make the passenger service decision making more demand responsive has stood under the spotlight. Therefore, as a base of the railway passenger service decision making, it is of paramount importance to study the railway passenger path choice.

Generally, it is assumed that railway passengers will choose their paths prior to their departure and then strictly follow their paths till they reach their destinations. More and more different train types are available for travelers; many paths need to be chosen by travelers from their departure to their destination. However, most travelers could not choose accurately depending on their travel demand before departure. Thus, special attention should be paid to the generalized cost of different paths in modeling railway

passenger's optimal path choice, which serves as a useful planning tool at both strategic and operational levels.

Rational railway passenger travel network is backbone of passenger optimal path choice. In most previous studies, a schedule-based transfer network was constructed by three kinds of nodes, which are station nodes, train arrival nodes, train departure nodes, and five kinds of arcs, which are boarding arcs, alighting arcs, running arcs, stopping arcs, and transferring arcs. And they defined the generalized cost function depends on a variety of contributing factors, including three items: travel time, fare, and discomfort cost (see [1]).

In one of the most influential papers on schedule-based transit modeling, Tong and Richardson [2] argued that a development algorithm for finding the time-dependent minimum path between two stations in a schedule-based transit system. Afterwards they proposed a recursive algorithm for finding the reasonable paths in a transit network (see [3]) and proposed a specially developed branch and bound algorithm, which was used to generate the time-dependent minimum

path for passenger flow assignment. The assignment procedure was conducted over a period in which both passenger demand and train headway are varying. They presented an overview of the research that had been carried out to develop the schedule-based transit assignment model (see [4, 5]). In addition, more schedule-based path-finding algorithms were presented for the transit network, such as forward-search algorithm and backward-search algorithm (see [6–8]). However, from empirical studies it appears clear that the vehicle capacity constraints were not considered. Thus, in order to have a more realistic description of technological progress, the concept of strategies in schedule-based transit networks with capacity constraints was extended in a study (see [9, 10]).

As it is described above, railway passengers will choose the path by the minimum generalized cost prior to their departure and then strictly follow their paths till they reach their destinations. Generally, a passenger will not transfer to another train in travel, if the previous train is congested or exceedingly uncomfortable. Our aim is to determine the shortest/optimal paths and depict them to the passenger for decision making before the journey. The developed algorithm based on residual train capacity does not inform the passenger which path is congested in travel; however, it can provide the passenger choices to make a decision on which path to traverse for completing his/her journey.

In the transportation field, many researchers presented the various path searching algorithms for different types of networks to solve shortest paths problem [11–14]. In [15] a multicriteria shortest path searching approach was presented based on the classical shortest path problem on a network representing the urban multimodal transportation system, to aim at minimizing the overall cost, time, and users' discommodity associated with the required paths. Authors in [16] presented a new route set generation algorithm based on shortest path search with link elimination. The proposed procedure combines a Breadth First Search with a topologically equivalent network reduction and ensures a high diversity between the routes. They demonstrated the usability of the algorithm, whose performance and resulting route sets were compared with those of a stochastic choice set generation algorithm.

The method widely applied in shortest path searching problem is Dijkstra's algorithm (see [17–19]). Recent studies have focused on optimized Dijkstra's algorithm combined with other methods to search the shortest path. Pradhan and Mahinthakumar used Floyd-Warshall algorithm and Dijkstra's algorithm to search the all-pairs shortest path in a large-scale transportation network [20]. Szűcs used the Dempster-Shafer theory and Dijkstra's algorithm to search the best route planning in a transport network where the network type can be arbitrary: a network of bus routes, a network of tram rails, a road network, or any other type of a transport network [21]. Namkoong et al. presented the optimal path searching algorithm which was basically based on Dijkstra's algorithm and the tree-building algorithm used to construct optimal paths in developing route guidance systems and traffic control systems [22]. In [23, 24] the methods is based on multicosts constraints and time-window

constraints, followed by the application of a K shortest path algorithm to find out the first K shortest paths in a schedule-based transit network. According to previous studies, we proposed an optimal path generation algorithm of railway passenger travel network based on residual train capacity and Dijkstra's algorithm to solve the optimal path searching problem.

The paper is organized as follows. In the next section, we construct the railway passenger travel network and present the notations used in the development of this network. The generalized cost function and residual capacity of the path as the foundation of path searching procedure are developed, and network topology is analyzed. The construction method of the railway passenger travel network on the basis of the train timetable is proposed. In Section 3, the optimal paths searching algorithm based on residual train capacity is presented. An illustrative example is presented to demonstrate the validity of the passenger travel network construction method and the optimal path searching algorithm in Section 4. In the last section, we present the conclusions and recommendations for future research.

2. Railway Passenger Travel Network Construction

2.1. Notation. We have the following notations

$v_{city,tr}^{s,ar}$ or $v_{city,tr}^{s,de}$: the node, which includes attributes as follow: station (s), node type (ar/de), city ($city$), and train number (tr),

cid : the code of $city$,

sid : the code of station s ,

tid : the code of train tr ,

$vtype$: the type of node, which consists of arrival node and departure node,

$vtime$: time of the train arrival node $v_{city,tr}^{s,ar}$ or time of the train departure node $v_{city,tr}^{s,de}$,

E_1, E_2, E_3, E_4, E_5 : the set of boarding arcs, the set of running arcs, the set of stopping arcs, the set of transferring arcs, and the set of alighting arcs, respectively,

V_1 : the set of the origin node v_O and the destination node v_D ,

V_2 : the set of the train arrival nodes and train departure nodes,

$(v_O, v_{city,tr}^{s,de}) \in E_1$: the arc between the origin node v_O and the train departure node $v_{city,tr}^{s,de}$ for the train tr at station s in city,

$(v_{city1,tr}^{s,de}, v_{city2,tr}^{s,ar}) \in E_2$: the arc between the train departure node $v_{city1,tr}^{s,de}$ for the train tr at station s in $city 1$ and the train arrival node $v_{city2,tr}^{s,ar}$ for the train tr at station s in $city 2$,

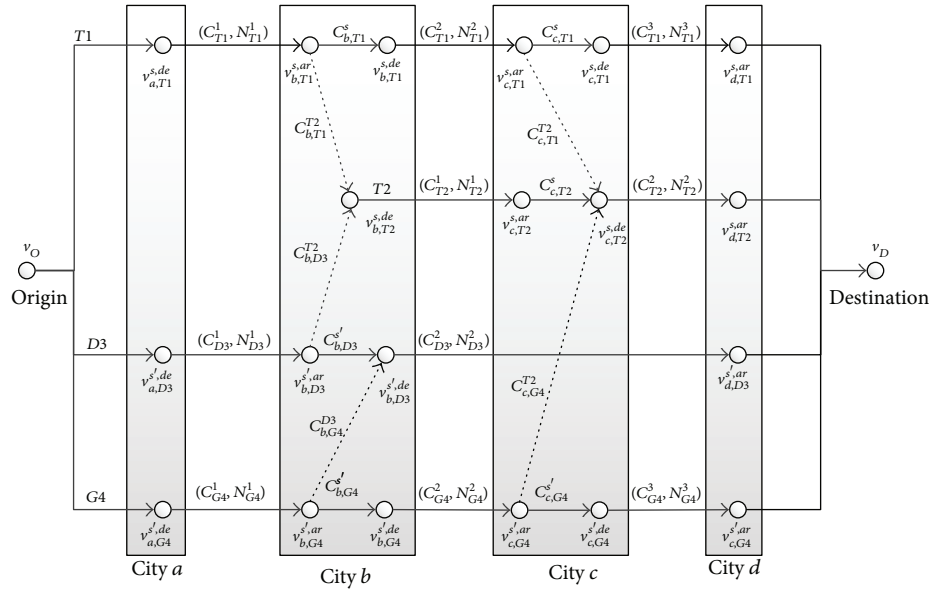


FIGURE 1: Railway passenger travel network.

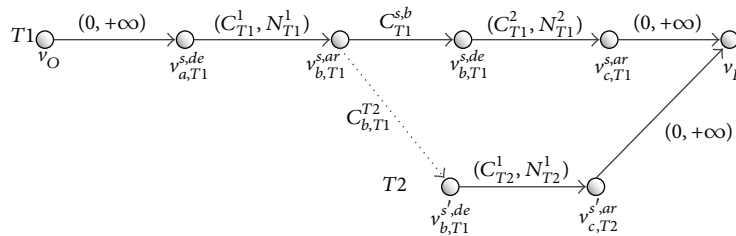


FIGURE 2: Two kinds of nodes and five kinds of arcs in railway passenger travel network.

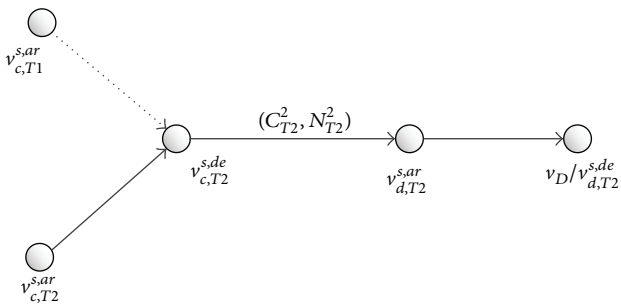


FIGURE 3: Network topology in condition that the train has residual capacity.

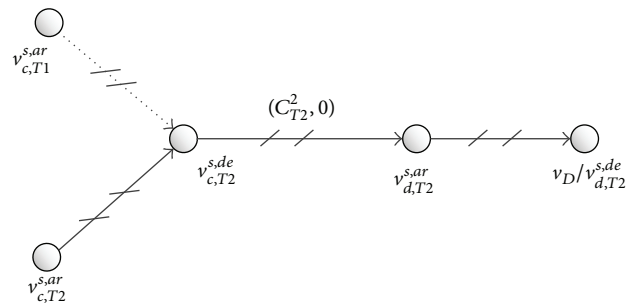


FIGURE 4: Network topology of disabled running arc.

$(v_{city,tr}^{s,ar}, v_{city,tr}^{s,de}) \in E_3$: the arc between the train arrival node $v_{city,tr}^{s,ar}$ for the train tr at station s in city 1 and the train departure node $v_{city,tr}^{s,de}$ for the train tr at station s in city 2,

$(v_{city,tr1}^{s,ar}, v_{city,tr2}^{s,de}) \in E_4$: the arc between the train arrival node $v_{city,tr1}^{s,ar}$ of the train $tr1$ and the train departure node $v_{city,tr2}^{s,de}$ of the transfer train $tr2$ at station s in city,

$(v_{city,tr}^{s,ar}, v_D) \in E_5$: the arc between the train arrival node $v_{city,tr}^{s,ar}$ for the train tr at station s in city and the destination node v_D ,

t_{arc} : the travel time of the arc,

f_{arc} : the travel fare of the arc,

eid : code of the arc,

$T[i][j]$: the travel time from node i to node j ,

$N[i][j]$: residual train capacity from node i to node j ,

$F[i][j]$: the travel fare from node i to node j ,

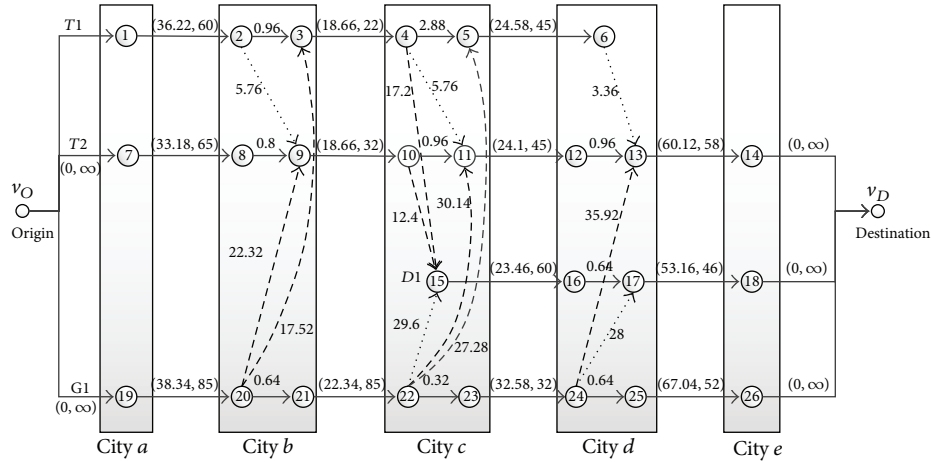


FIGURE 5: Railway passenger travel network based on railway timetables in the example.

$C[i][j]$: the generalized cost from node i to node j ,

Maxnum: a maximum value.

2.2. Railway Passenger Travel Network. Railway passenger travel network $G(V, E)$ is constructed based on the schedule-based transfer network by two kinds of nodes and five kinds of arcs, where V and E are the set of nodes and the set of arcs, respectively. Define the set of nodes as $V = V_1 \cup V_2$ and the set of arcs as $E = E_1 \cup E_2 \cup E_3 \cup E_4 \cup E_5$. There are two kinds of nodes in this network, one kind of nodes is transition node which is original node v_O or destination node v_D . The other kind of nodes is train arrival nodes or train departure nodes. There are five kinds of arcs in this network: the boarding arcs, running arcs, stopping arcs, transferring arcs, and alighting arcs. To illustrate, Figure 1 presents a railway passenger travel network with four cities and four trains. Clearly, a passenger travel path from the origin node to the destination node includes a series of boarding, running, stopping, transferring, and alighting arcs in the passenger travel network (V, E) .

In Figure 2, there are two paths which are $P1$ and $P2$ in the passenger travel network with three cities (city a , city b , and city c) and two trains (train $T1$ and train $T2$). Path $P1 = \{(v_O, v_{a,T1}^{s,de}), (v_{a,T1}^{s,de}, v_{b,T1}^{s,ar}), (v_{b,T1}^{s,ar}, v_{c,T1}^{s,de}), (v_{c,T1}^{s,de}, v_D)\}$, which includes the boarding arc $(v_O, v_{a,T1}^{s,de})$, running arcs $(v_{a,T1}^{s,de}, v_{b,T1}^{s,ar}) \cup (v_{b,T1}^{s,ar}, v_{c,T1}^{s,de})$, stopping arc $(v_{b,T1}^{s,ar}, v_{c,T1}^{s,de})$, and alighting arc $(v_{c,T1}^{s,de}, v_D)$. Path $P2 = \{(v_O, v_{a,T1}^{s,de}), (v_{a,T1}^{s,de}, v_{b,T1}^{s,ar}), (v_{b,T1}^{s,ar}, v_{b,T2}^{s',de}), (v_{b,T2}^{s',de}, v_{c,T2}^{s',ar}), (v_{c,T2}^{s',ar}, v_D)\}$, which includes the boarding arc $(v_O, v_{a,T1}^{s,de})$, running arcs $(v_{a,T1}^{s,de}, v_{b,T1}^{s,ar}) \cup (v_{b,T2}^{s',de}, v_{c,T2}^{s',ar})$, transferring arc $(v_{b,T1}^{s,ar}, v_{b,T2}^{s',de})$, and alighting arc $(v_{c,T2}^{s',ar}, v_D)$.

2.3. The Generalized Cost Function. As it is described in Section 1, a passenger will not transfer to another train in travel, even if the previous train is congested or exceedingly uncomfortable. Our aim is to determine the shortest/optimal paths and depict them to the passenger for decision making

before the journey. Therefore, the most important factors of passenger travel are the travel time and travel fare of the path. The discomfort cost is not informed of in the developed algorithm. Before the journey, a passenger can decide travel path by considering whether the fare or time of the path the passenger would choice is smaller than that of other optional paths. We define this cost as the generalized cost function that can be represented as

$$C_{arc} = \alpha \cdot t_f \cdot t_{arc} + \beta \cdot f_{arc}, \quad (1)$$

where t_f represents average values of time for the set of arcs and α, β denote, respectively, time parameter and fare parameter, $\alpha + \beta = 1$.

As an example, consider the running arc $(v_{a,T1}^{s,de}, v_{b,T1}^{s,ar})$ as shown in Figure 1. Assume that the time of the departure node is 17:08 and the time of the arrival node is 18:34 and that the fare of this running arc is 24.5 CNY. Then, we can calculate the generalized cost of the running arc based on (1) as

$$C_{arc} = \alpha \times t_f \times (18 : 34 - 17 : 08) + \beta \times 24.5. \quad (2)$$

In the railway passenger travel network, the generalized cost of the path is the sum of all arcs belonging to this path. Define the arc weight (C, N) , where C and N represent the generalized cost and the residual capacity of arc, respectively. In particular, define the residual capacity of the boarding arc, stopping arc, transferring arc, and alighting arc $N = \infty$, the generalized cost of boarding arc and alighting arc $C = 0$, and the fare of the stopping arc $f = 0$. The minimal capacity of all arcs in a path is the capacity of this path.

2.4. Network Topology Analysis Based on Residual Train Capacity. Residual train capacity is declining when more and more passengers travel in the passenger travel network (Figure 3). In particular, the running arc of a train cannot transport more passengers when the train has no residual capacity; in other words, the running arc is disabled. Then the passenger travel network is changed. In Figure 4, the running arc $(v_{c,T2}^{de}, v_{d,T2}^{ar})$ is disabled when the train cannot transport

TABLE 1: Standard train timetable.

City	<i>a</i>				<i>b</i>		<i>c</i>			
Station	<i>a1</i>		<i>a2</i>		<i>b</i>		<i>c1</i>		<i>c2</i>	
Type	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure
Train										
<i>T1</i>	—	—	—	17:08	18:34	18:52	20:35	—	—	—
<i>T2</i>	—	15:00	—	—	17:33	17:38	—	—	20:50	20:56

TABLE 2: Railway timetable.

City	<i>a</i>		<i>b</i>		<i>c</i>		<i>d</i>		<i>e</i>	
Train	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure	Arrival	Departure
<i>T1</i>	—	14:10	17:02	17:08	18:34	18:52	20:35	—	—	—
<i>T2</i>	—	15:00	17:33	17:38	19:04	19:10	20:50	20:56	01:38	—
<i>D1</i>	—	—	—	—	—	19:44	20:40	20:44	22:50	—
<i>G1</i>	—	14:37	15:56	16:00	16:39	16:41	17:49	17:53	19:47	—

more passengers; then the departure node $v_{c,T2}^{de}$ and the arrival node $v_{d,T2}^{ar}$ are disabled. According to network features, the stopping arc ($v_{c,T2}^{ar}, v_{c,T2}^{de}$) and the transferring arc ($v_{c,T1}^{ar}, v_{c,T2}^{de}$) which connected to the node $v_{c,T2}^{de}$ are disabled. Similarly, the stopping arc ($v_{d,T2}^{ar}, v_{d,T2}^{de}$) and the alighting arc ($v_{d,T2}^{ar}, v_D$) which connected to the node $v_{d,T2}^{ar}$ are disabled.

2.5. *Construction Algorithm of Passenger Travel Network Based on Train Timetable.* The passenger travel network construction algorithm is shown in Algorithm 1.

3. Traveling Optimal Path Searching Algorithm

3.1. *Analysis of the Optimal Paths Searching Problems.* In this paper, we consider the following optimal path searching problem of railway passenger travel network: passenger flow guidance and passenger flow equilibrium distribution.

With regard to passenger flow guidance problem, passengers, whose travel time is definite, choose their optimal paths prior to their departure and then strictly follow their paths till they reach their destinations. They consider actual travel time, travel fare, or generalized cost in the actual train timetable (Table 1).

With regard to passenger flow equilibrium distribution problem, it refers to two situations including passenger flow distribution in emergence scenarios and simulation analysis of passenger flow distribution. In emergency scenarios, all stranded passengers need to be reasonably and efficiently guided to a safe place (another city). As for normal scenarios, the simulated passenger flow distribution is also required for the analogy and analysis of the practical passenger flow distribution.

Therefore, a general, feasible, and flexible method which could solve both the problems we revealed above is expected.

3.2. *Traveling Optimal Path Searching Algorithm (TOPSA).* Search the optimal path by Dijkstra’s algorithm and then

assign passenger flow to this path according to the capacity of the optimal path until a running arc of this path is disabled. A new passenger travel network is formed after the disable arc is deleted. Continue to search the optimal path and assign the remaining passengers until all passengers are assigned or all paths have no residual capacity. TOPSA is shown in Algorithm 2.

3.3. *Computational Complexity of TOPSA.* Generally, the upper bound of the computational complexity of optimal paths search algorithm (Dijkstra’s algorithm) from origin node to destination node is $O(n^2)$, where n represents total number of nodes in the travel network. In TOPSA, in this paper, the algorithm stops when the remaining passengers $P - \sum_k N_{\min}(l_k) \leq 0$. Therefore, the computational complexity of TOPSA is $O(k \cdot n^2)$, where k represents total number of optimal paths. In addition, the total number of optimal paths is limited by the size of the travel network and residual capacity; k ($k \ll n$) is generally single digit. Thus, the computational complexity of TOPSA is $O(n^2)$.

4. Numerical Example

The proposed approach is illustrated with a railway timetable as shown in Table 2. Four trains run among cities $a, b, c, d,$ and e . Residual capacities of all trains are shown in Table 3. The minimal train transfer times of the same station and different station in a city are 15 min and 30 min, respectively. Use the proposed approach to construct railway passenger travel network based on the railway timetable (Table 2), as shown in Figure 5. Suppose all the fare of transferring arcs among different stations $f_{\text{tar}} = 30$ CNY, the value of time $t_f = 12$ CNY/h, and the parameters $\alpha = 0.8$ and $\beta = 0.2$.

In the travel network, suppose that 100 travelers need to travel from origin node of the city a to the destination node of the city e . The fare and capacity of running arcs are as shown in the Table 3. According to the method introduced in Section 3, four paths are found and passenger flow is assigned

TABLE 3: Fare (unit: CNY) and capacity of the running arcs among cities.

Arc	$a \rightarrow b$		$b \rightarrow c$		$c \rightarrow d$		$d \rightarrow e$	
	Fare	Capacity	Fare	Capacity	Fare	Capacity	Fare	Capacity
<i>T1</i>	43.5	60	24.5	22	40.5	45	—	—
<i>T2</i>	43.5	65	24.5	32	40.5	45	75	58
<i>D1</i>	—	—	—	—	72.5	60	165	46
<i>G1</i>	128.5	85	80.5	85	108.5	32	244	52

TABLE 4: The optimal paths information for 100 travelers.

	Path	Generalized cost	Passenger flow
<i>P1</i>	$v_o \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow v_D$	138.78	32
<i>P2</i>	$v_o \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 13 \rightarrow 14 \rightarrow v_D$	146.78	22
<i>P3</i>	$v_o \rightarrow 19 \rightarrow 20 \rightarrow 21 \rightarrow 22 \rightarrow 23 \rightarrow 24 \rightarrow 25 \rightarrow 26 \rightarrow v_D$	161.90	32
<i>P4</i>	$v_o \rightarrow 19 \rightarrow 20 \rightarrow 21 \rightarrow 22 \rightarrow 15 \rightarrow 16 \rightarrow 17 \rightarrow 18 \rightarrow v_D$	168.18	14

TABLE 5: The optimal paths information for 250 travelers.

	Path	Generalized cost	Passenger flow
<i>P1</i>	$v_o \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow v_D$	138.78	32
<i>P2</i>	$v_o \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 13 \rightarrow 14 \rightarrow v_D$	146.78	22
<i>P3</i>	$v_o \rightarrow 19 \rightarrow 20 \rightarrow 21 \rightarrow 22 \rightarrow 23 \rightarrow 24 \rightarrow 25 \rightarrow 26 \rightarrow v_D$	161.90	32
<i>P4</i>	$v_o \rightarrow 19 \rightarrow 20 \rightarrow 21 \rightarrow 22 \rightarrow 15 \rightarrow 16 \rightarrow 17 \rightarrow 18 \rightarrow v_D$	168.18	46
<i>P5</i>	$v_o \rightarrow 19 \rightarrow 20 \rightarrow 21 \rightarrow 22 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow v_D$	176.64	4

Input: Standard train timetable (Table 1), minimum transferring time $K[sid(i)][sid(j)]$, transferring fare $M[sid(i)][sid(j)]$ and parameters α, β .

Output: $G(V, E)$, $V(vid, cid, sid, tid, vtype, vtime)$ and $E(eid, N[i][j], C[i][j])$.

Step 1. Initialize node set V , define total number of nodes n according to train timetable.

Step 2. Establish tid and the time among arcs.

for ($i = 1, i \leq n, i++$)

if $tid(i) = tid(i + 1)$, set $T[i][i + 1] \leftarrow vtime(i + 1) - vtime(i)$.

Step 3. Establish all transferring arcs

for ($k = 1, k \leq \text{amount of cities}, k++$)

for ($j = 1, j \leq n, j++$)

if ($cid(j) = k \ \&\& \ vtype(j) = "de"$)

for ($i = 1, i \leq n, i++$)

if ($cid(i) = k \ \&\& \ vtype(i) = "ar" \ \&\& \ (tid(i) \neq tid(j))$)

set $p \leftarrow vtime(j) - vtime(i)$

if ($p \geq K[sid(i)][sid(j)]$)

set $N[i][j] \leftarrow \text{maxnum}, T[i][j] \leftarrow p, F[i][j] \leftarrow M[sid(i)][sid(j)],$

$C[i][j] \leftarrow \alpha T[i][j] + \beta F[i][j];$

otherwise $N[i][j] \leftarrow 0, T[i][j] \leftarrow \text{maxnum}.$

Step 4. Set capacity of all stopping arcs.

for ($i = 1, i \leq n, i++$)

if $vtype(i) = "ar"$, set $N[i][i + 1] \leftarrow \text{maxnum}.$

Step 5. Set capacity and fare of all running arcs.

for ($i = 1, i \leq n, i++$)

if $vtype(i) = "de"$

set $N[i][i + 1], C[i][i + 1] \leftarrow \alpha T[i][i + 1] + \beta F[i][i + 1];$

if $N[i][i + 1] = 0$, set $C[i][j] \leftarrow \text{maxnum}; \text{end}.$

Input: $G(V, E)$, origin city x , destination city y , departure time T_0 , total number of passengers P .
Output: The set of the path $L(l_1, l_2, \dots, l_k)$, the generalized cost $C(l_k)$ of path l_k and passenger volume $P(l_k)$ of path l_k .
Step 1. Set $V_o(vid(O) = 0, cid(O) = x, vtype = "O")$, $V_d(vid(D) = n + 1, cid(D) = y, vtype = "D")$.
for ($j = 1, j \leq n, j++$)
if ($cid(j) = x \ \&\& \ vtype(j) = "de" \ \&\& \ vtime > T_0$)
 $N[0][j] \leftarrow \text{Maxnum}, C[0][j] \leftarrow 0$;
for ($i = 1, i \leq n, i++$)
if ($cid(i) = y \ \&\& \ vtype(i) = "ar"$)
 $N[i][n + 1] \leftarrow \text{Maxnum}, C[i][n + 1] \leftarrow 0$;
set $k = 1$;
Step 2. From origin node O , use Dijkstra algorithm to search the optimal path l_k according to the generalized cost of the arc.
And calculate the generalized cost $C(l_k)$ of this path.
if $C(l_k) \geq \text{Maxnum}$, delete l_k , end;
otherwise $l_k: V_o \rightarrow V_i \rightarrow \dots \rightarrow V_j \rightarrow V_d$ is deposited to the set L .
Step 3. Sort the arcs by residual train capacity in path l_k , set the minimal arc capacity is capacity of the path l_k , namely $N_{\min}(l_k)$; set passenger volume of path $l_k: P(l_k) = N_{\min}(l_k)$;
set $N[i][j] \leftarrow N[i][j] - N_{\min}(l_k), P \leftarrow P - N_{\min}(l_k)$; if the arc in the path l_k
 $N[p][q] = 0$, set $C[p][q] \leftarrow \text{Maxnum}$.
Step 4. Reconstruction of the passenger travel network:
set $C[i][p] \leftarrow \text{Maxnum}, C[q][j] \leftarrow \text{Maxnum}$;
if $P \leq 0$, set $P(l_k) \leftarrow P + P(l_k)$, end;
otherwise $k \leftarrow k + 1$, go to Step 2.

ALGORITHM 2

in the travel network, which are shown in Table 4, costing 0.46 s on the computer with an E7000+CPU and 2 G RAM.

Obviously, everyone hopes to choose the optimal/shortest path PI for his/her journey. However, in fact, the residual capacity of the train TI is not enough and only 32 travelers are distributed in the first path PI . Secondary optimal paths will be searched in a row and the remaining passengers will be distributed, respectively, until everyone can get on the train.

The number of passengers increases from 100 to the maximum capacity of trains, the optimal path is still searched, and the remaining passengers are assigned until all passengers are assigned or all paths have no residual capacity. For example, suppose that 250 travelers need to travel from origin node of the city a to the destination node of the city e . The fare and capacity of running arcs are unchanged. According to the method introduced in Section 3, five paths are found and passengers are assigned in the travel network, which are shown in Table 5, costing 0.50 s on the computer with an E7000+CPU and 2 G RAM. Only 136 travelers are assigned, and others cannot go to destination because all paths have no residual capacity in this travel network.

5. Conclusion

Railway transport enjoys great and increasing popularity as an essential transport mode for medium-long distance journey in many countries in the past decades. However, in condition when more and more different train types are available, railway passengers are often perplexed to choose the best travel plan. This study developed a rational approach to construct the railway passenger travel network, designed the generalized cost function on the path as the foundation of

path searching procedure, and analyzed the network topology. In particular, we proposed an optimal path searching algorithm which combined the Dijkstra algorithm and the residual train capacity based on the railway passenger travel network, and the validity of the proposed approach was verified with an illustrative example. For future work, one possible extension of this research is to consider the transfer reliability and the path choice behaviors of various passenger groups. The proposed approach can push the construction of optimal train schedule and will be the theoretical foundation in decision making of resource allocation. In a more general sense of application, the proposed method could be adapted in multimodal transportation systems especially in metro transport.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work has been supported by National Natural Science Foundation of China (61074151), National Key Technology Research and Development Program of China (2009BAG12A10), and National High Technology Research and Development Program 863 of China (2012AA112001).

References

- [1] F. Shi, Z. Zhou, J. Yao, and H. Huang, "Incorporating transfer reliability into equilibrium analysis of railway passenger flow,"

- European Journal of Operational Research*, vol. 220, no. 2, pp. 378–385, 2012.
- [2] C. O. Tong and A. J. Richardson, “A computer model for finding the time dependent minimum path in transit systems with fixed schedules,” *Journal of Advanced Transportation*, vol. 18, no. 2, pp. 145–161, 1984.
- [3] C. O. Tong and S. C. Wong, “An algorithm for finding reasonable paths in transit networks,” *Transportation Research B*, vol. 33, pp. 107–121, 1999.
- [4] C. O. Tong, S. C. Wong, M. H. Poon, and M. C. Tan, “A schedule-based dynamic transit network model—recent advances and prospective future research,” *Journal of Advanced Transportation*, vol. 35, no. 2, pp. 175–195, 2001.
- [5] M. H. Poon, S. C. Wong, and C. O. Tong, “A stochastic transit assignment model using a dynamic schedule-based network,” *Transportation Research B*, vol. 38, pp. 343–368, 2004.
- [6] A. Nuzzolo, F. Russo, and U. Crisalli, “A doubly dynamic schedule-based assignment model for transit networks,” *Transportation Science*, vol. 35, no. 3, pp. 268–285, 2001.
- [7] R. H. Huang and Z. R. Peng, “Schedule-based path-finding algorithms for transit trip-planning systems,” *Transportation Research Record*, no. 1783, pp. 142–148, 2002.
- [8] R. Huang, “A schedule-based pathfinding algorithm for transit networks using pattern first search,” *GeoInformatica*, vol. 11, no. 2, pp. 269–285, 2007.
- [9] N. Rochau, K. Nökel, and M. G. H. Bell, “Cost functions for strategies in schedule-based transit networks with limited vehicle capacities,” *Transportation Research Record*, no. 2284, pp. 62–69, 2012.
- [10] A. Nuzzolo, U. Crisalli, and L. Rosati, “A schedule-based assignment model with explicit capacity constraints for congested transit networks,” *Transportation Research C: Emerging Technologies*, vol. 20, no. 1, pp. 16–33, 2012.
- [11] L. Fu, D. Sun, and L. R. Rilett, “Heuristic shortest path algorithms for transportation applications: state of the art,” *Computers and Operations Research*, vol. 33, no. 11, pp. 3324–3343, 2006.
- [12] N. Katoh, T. Ibaraki, and H. Mine, “An efficient algorithm for K shortest simple paths,” *Networks*, vol. 12, no. 4, pp. 411–427, 1982.
- [13] D. Eppstein, “Finding the k shortest paths,” *SIAM Journal on Computing*, vol. 28, no. 2, pp. 652–673, 1999.
- [14] S. Zhang and X. Liu, “A new algorithm for finding the k shortest transport paths in dynamic stochastic networks,” *Journal of Vibroengineering*, vol. 15, no. 2, pp. 726–735, 2013.
- [15] P. Modesti and A. Sciomachen, “A utility measure for finding multiobjective shortest paths in urban multimodal transportation networks,” *European Journal of Operational Research*, vol. 111, no. 3, pp. 495–508, 1998.
- [16] N. Rieser-Schuessler, M. Balmer, and K. W. Axhausen, “Route choice sets for very high-resolution data,” *Transportmetrica A-Transport Science*, vol. 9, no. 9, pp. 825–845, 2013.
- [17] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.
- [18] F. B. Zhan, “Three fastest shortest path algorithms on real road networks: data structures and procedures,” *Journal of Geographic Information and Decision Analysis*, vol. 1, no. 1, pp. 69–82, 1997.
- [19] D. van Vliet, “Improved shortest path algorithms for transport networks,” *Transportation Research*, vol. 12, no. 1, pp. 7–20, 1978.
- [20] A. Pradhan and G. Mahinthakumar, “Finding all-pairs shortest path for a large-scale transportation network using parallel Floyd-Warshall and parallel Dijkstra algorithms,” *Journal of Computing in Civil Engineering*, vol. 27, no. 3, pp. 263–273, 2013.
- [21] G. Szűcs, “Route planning based on uncertain information in transport networks,” *Transport*, vol. 27, no. 1, pp. 79–85, 2012.
- [22] S. Namkoong, J. H. Rho, and J. U. Choi, “Development of the tree-based link labeling algorithm for optimal path-finding in urban transportation networks,” *Mathematical and Computer Modelling*, vol. 27, no. 9–11, pp. 51–65, 1998.
- [23] W. T. Xu, S. W. He, R. Song, and S. S. Chaudhry, “Finding the K shortest paths in a schedule-based transit network,” *Computers and Operations Research*, vol. 39, no. 8, pp. 1812–1826, 2012.
- [24] D. Canca, A. Zarzo, P. L. González-R, E. Barrena, and E. Algaba, “A methodology for schedule-based paths recommendation in multimodal public transportation networks,” *Journal of Advanced Transportation*, vol. 47, no. 3, pp. 319–335, 2013.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

