

# A USER PREFERABLE K-SHORTEST PATH ALGORITHM FOR INTERMODAL NETWORK

Zhuo SUN  
PhD Student  
Graduate School of Environmental Studies  
Nagoya University  
Furo-cho Chikusa-ku Nagoya  
464-8603, Japan  
Fax: +81-52-789-3837  
E-mail: [zhuosun@urban.env.nagoya-u.ac.jp](mailto:zhuosun@urban.env.nagoya-u.ac.jp)

Qionghua WU  
PhD Student  
Graduate School of Environmental Studies  
Nagoya University  
Furo-cho Chikusa-ku Nagoya  
464-8603, Japan  
Fax: +81-52-789-3837  
E-mail: [wujoan@urban.env.nagoya-u.ac.jp](mailto:wujoan@urban.env.nagoya-u.ac.jp)

Hirokazu KATO  
Associate Professor  
Graduate School of Environmental Studies  
Nagoya University  
Furo-cho Chikusa-ku Nagoya  
464-8603, Japan  
Fax: +81-52-789-3837  
E-mail: [kato@genv.nagoya-u.ac.jp](mailto:kato@genv.nagoya-u.ac.jp)

Yoshitsugu HAYASHI  
Professor and Dean  
Graduate School of Environmental Studies  
Nagoya University  
Furo-cho Chikusa-ku Nagoya  
464-8603, Japan  
Fax: +81-52-789-3837  
E-mail: [yhayashi@genv.nagoya-u.ac.jp](mailto:yhayashi@genv.nagoya-u.ac.jp)

**Abstract:** This study discusses seamlessly integrating multiple modes of transportation networks and calculating the User Preferable K-Shortest paths in the network between origin and destination with a new algorithm based on improved Link Penalty. A projection method has been used to connect different networks. Three aspects of users' preferences have been evaluated to set the rate of link penalties in each step that k-shortest paths will be generated. Link similarities are considered respectively in different mode and Partial Overlap has been introduced to be a criterion for iteration. Users can set their constraints and will get very precise and efficient travel information through GIS. Due to the simplification of the topological structure and flexibility of the k-shortest path algorithm, it's very easy for developers to update their existing traveler information system.

**Key Words:** *K-Shortest Path, Intermodal Network, Link Penalty*

## 1. INTRODUCTION

### 1.1 Problems in Current Traveler Information Systems

With the rapid development of urban infrastructure, multiple modes of transportation systems connect every corner of the world. The multimodal transportation systems can offer people or goods many ways to travel from one place to another place. The appropriate use of the intermodal transportation network can reduce traffic demands, relief traffic congestion and air pollution. Thanks to today's information technology, people can access online travel information wherever they are. Even a cell phone can provide a GIS map to people and guide their movement. Those systems that help people to make plans for trips are called Traveler Information System (TIS). It is essential for TIS to offer travelers the accurate and reliable travel information such as the needed time, cost, and transfers. Current TISs run on servers and are supported by databases which can get real time information of different transportation networks. They could provide users the best travel plan consisted of a path that has minimum cost of the whole trip. Some time users also want to know other alternative paths beside the best one. The reason is there are many uncertainties in real world and the system should provide more choices to users to make them decide in their own situation. Current TISs can

only provide users some alternative paths with redundant information. For example, if a user wants to go to a specified place in another city, current TISs will give him a list of travel plans consist of transfers in subway and train. If the subway system is complex most of these plans have many comparisons in subway but not distinct comparisons of different trains which is the most important thing a user concerns. These diverse travel plans that confused users are not user preferable results. There are three aspects that people always concern for travels: time, cost and convenience. The users' preferences should be considered and maximized in travel plans but most of TISs just enumerate all of the possible paths without considering different transportation modes. The limitation which current TISs have is rooted in the k-shortest path algorithm.

### **1.2 The K-Shortest Path Algorithm**

The shortest path algorithm was first introduced to solve the optimization problem of traveling from an origin to a destination. In some cases, the shortest path is not adequate and some alternatives, like the 2<sup>nd</sup>, the 3<sup>rd</sup>... the k<sup>th</sup> shortest path, need to be found beside the shortest one. Yen (1971) first introduced a k-shortest path searching method by deleting node from the network, and then several k-shortest algorithms have been suggested. There are two groups of k-shortest path algorithm. The difference is one allow loops in result paths and another don't. In transportation network the latter is always used. The k-shortest path algorithms have been used widely in current TISs for generating different travel plans. Although k-shortest path algorithm can provide several alternative paths, it has inherent limit of heavy overlapping among derived paths, which may lead to redundant travel information to the users. There are many researches published regarding this problem. Barra *et al.* (1993) proposed a link penalty method such that the network is modified by increasing the cost of all links on the shortest path. After modifying the network a new shortest path is computed according to the increased costs and the process continues until no more paths are required or no more paths can be determined. Scott *et al.* (1997) proposed an approach that would find the best k-similar paths that have at most k links in common with the shortest path. Akgun *et al.* (2000) determined a dissimilar path set by measuring the spatial dissimilarity between any two paths in the path set. The dissimilar paths are calculated by choosing some paths from the paths set in a way that the minimum of repeated links between any two paths is maximized. Lim *et al.* (2005) introduce a modified method of Barra that measure the overlap of the k-shortest paths as the similarity criterion.

### **1.3 The User Preferable K-Shortest Path Algorithm**

Although to find dissimilar paths is very important for the k-shortest path algorithm, in intermodal transportation network a path from an origin to a destination may contains many minor links whose dissimilarity will not impact the whole path but confuse users. An efficient k-shortest path algorithm for intermodal transportation network should have dissimilar paths in major mode and similar paths in minor mode. This kind of k-shortest path algorithm is named "User Preferable K-Shortest Path Algorithm" by this study.

The purpose of this study is to develop a user preferable k-shortest path algorithm for intermodal transportation network. A projection method which described by Sun *et al.* (2005) will be used to connect different networks. Users' preferences could be evaluated in different mode to set the rate of link penalties in each step that k-shortest paths will be generated. A new method named partial overlap has been developed based on Lim's path overlap method to control the iteration. Follow this chapter; the method that builds the intermodal transportation network will be described. Then the user preferable k-shortest path algorithm will be discussed in detail. Finally a numerical example will be given.

## 2. INTEGRATE MULTIPLE MODES OF NETWORKS

### 2.1 The Construction of Mode 0, 1, 2 ... Network in GIS

The transportation networks could be classified by the mode of transport, such as human-powered transport (Cyclist, Pedestrian), road transport (car, bus), rail transport (subway, train) and etc. Every mode of transport is serving for people. The human-powered transport could be set as the base mode because all other modes consequently connect this mode for transporting people or goods. Thus the street or road transportation network could be set as base mode of network. Here mode 0 will be denoted as the base mode.

To construct the mode 0 or road network in GIS there are several steps. First, the road layer is created based on raster images, above which the topological structure of road network is built. The attributes of road links is calculated by using spatial function of GIS and stored in the road network directly. The item of Length can be gained through the length of the vector “polyline” in GIS and then divided by the scale of the raster image. The length unit is set to kilometer. The item of Travel Time can be gained through the average walking speed (5km/hour) and the length.

The other modes of transportation networks could be created in other layers depending on the road network or separately (Figure 1). For example, the transit network is stored on an independent layer over the road layer. Transit lines can be created by referring to roads lines. Then stops can be built on the lines according to their real location, and then the lines are split by the created stops. Finally connect lines segments and stops to form a transit network. The data structure of links in the transit network is similar to the links in road network except for the item of Travel Time. It can be calculated according to the real road property and the transit mode. For example, there is a bus route in the network. The route’s Travel Time can be obtained based on bus speed (about 30km/hour as usual) and the route length. If the ticket price increases along with the increasing of mileage during transit, the ticket price can be stored in Cost field or converted to equivalent time by using an appropriate formula. If there are different ticket prices for same journey and same bus line, to handle the multiple choices the single bus line could be separated to multiple lines according to ticket prices.

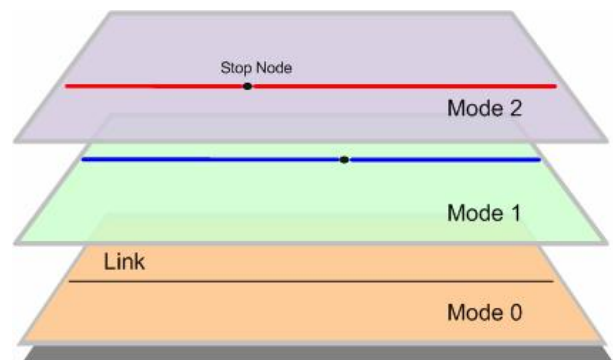


Figure 1 Build Multiple Modes of Networks

### 2.2 The Combination of Each Network

After each mode of network is built, they need to be integrated in order to perform further calculation and analysis. The topological structure used in this study is from Sun *et al.* (2005) in which a projection method had been introduced. The stop nodes are projected to road network in GIS and two dummy links to link every stop node and projected node (Figure 2). In this way integrated network shows a very simple and

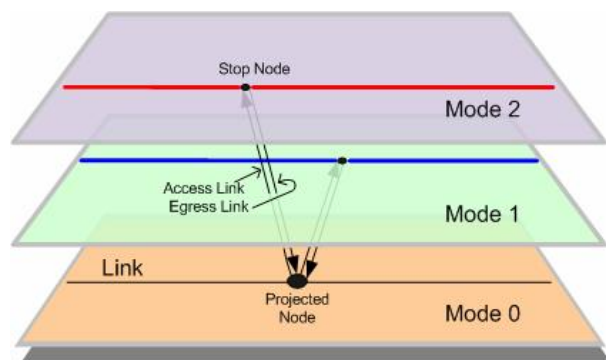


Figure 2 the Projection Method

accurate structure to be calculated and modified. Network impedances, such as cost, distance, transfer, waiting time and traveling time could be directly stored on corresponding links and existing shortest and k-shortest path algorithms can be applied without any modification.

The Travel Time and Cost fields in access links can be set to dwelling time and ticket price. Those fields in egress links can be set to dwelling time and 0. If the cost needs to be converted to equivalent time, formulas are listed as follows:

$$Ta = t_w + t_b + w \times t_c, Te = t_d \quad (1)$$

$$t_c = \frac{\text{ticket price} \times \text{legal annual working time}}{\text{annual income per head}} \quad (2)$$

Where  $Ta$  is the value of Travel Time of an access link,  $t_w$  represents waiting time,  $t_b$  represents the required time spent on getting on a vehicle from the outside of a stop.  $t_c$  represents the time converted from the ticket price.  $w$  is weight used to adjust the percentage of time after conversion.  $Te$  presents Travel Time of egress links and it is equal to  $t_d$ , which is the time one spend getting off vehicles plus getting out of the stop.

There are some points that should be noticed when forming the topological structure:

1. If many modes of networks have their stops at same location, these stops cannot be projected into one node on the link of mode 0 network. That will reduce the fragment in this area and reduce network complexity.
2. All modes of network should be converted to one-way link to match the “bigraph” topology. In practice, a two-way road or route can be presented by two directed links which are parallel and have opposite direction. After that the first principle also should be considered for setting up stops on each link.
3. The impedances  $Ta$ ,  $Te$  and  $cost$  etc. could be dynamic if congestion is taken into account. But in practice it's difficult to evaluate the congestion at the boarding and alighting point. The flow of people should be calculated.
4. If there are different ticket prices for same journey, the link in this mode can be separated in several parallel links according to the number of prices. But a more applicable method is reading the ticket prices dynamically at run time.
5. It is not always necessary to calculate the generalized time. It depends on the objective of travelers. In most cases only time or cost needs to be calculated because the generalized time will puzzle travelers.

### 3. THE USER PREFERABLE K-SORTEST PATH ALGORITHM

The multiple modes of transportation networks have been integrated by using above method. This intermodal network has a uniform structure and a shortest path from origin to destination can be calculated with a standard shortest path algorithm. This shortest path always can not satisfy people. Then a k-shortest path algorithm should be considered to generate more alternatives other than the shortest one. Because everything is severing for people, the people's preferences should be put in the first place when performing a k-shortest path algorithm.

#### 3.1 The User Preferences and Constraints

All of user preferences in finding a travel plan can be attributed to four objectives: the shortest time, the minimum cost, the maximum convenience and the maximum comfort. The time and cost in a trip is related to link sequences. The convenience that users concern in a trip is always related to the number of transfer or the walking distance. The comfort is hard to

evaluate but it is somewhat proportional to the cost. So the comfort could be removed from the list of the user preferences. Now there are three of user preferences left. These three objectives are hard to be reached simultaneously. Therefore only one objective or a compromising objective which consist of a proportion of each objective will be set. The shortest time and the shortest cost could be set as objective respectively or be set together as one object by multiply a factor to each one. The convenience which always be presented by the number of transfers also can set as an objective but users usually prefer giving it a range but not a specific value. That could be treated as a constraint.

As mentioned above finally the user preferences include two major objectives and some constraints. There is also another hidden user preference exists in the whole process. Users always want results path has dissimilarity in major mode transport and similarity in minor mode transport. If we denote  $P(o,d)$  as a path from node  $o$  to  $d$  in a intermodal network  $N$ ,  $I(P(o,d))$  as the total impedance of  $P(o,d)$ ,  $T(P(o,d))$  as the transfer number or the number of mode change(They are not strictly treated as two things) in  $P(o,d)$ ,  $M(P(o,d))$  as the major link in  $P(o,d)$ ,  $D(M(P(o,d)))$  as the dissimilarity indicator of the major link in link set (in this study the partial overlap is used), then we have the objective of the model,

$$\max : D(M(P(o,d))) \quad (3)$$

$$s.t. I(P(o,d)) < A \quad (4)$$

, and the objective of the submodel,

$$\min : I(P(o,d)) \quad (5)$$

$$s.t. T(P(o,d)) < B \quad (6)$$

, where  $A$  and  $B$  are constants.

This optimization problem is very hard to solve with traditional methods. In practice the submodel can be solved by dijkstra algorithm and the top level model can be solved by proposed algorithm in this study that will be described later. Note that the objective function above just presents the problem schematically. Many constraints will be added in a real problem, such as the walking distance, the waiting time, etc. They have been hidden above for better understanding but will be considered if travelers specify them.

### 3.2 Preparation of Data and Weights

It is necessary to build the data structure in computer memory before calculating. Since the final merged structure of this research is only a simple bigraph, the process of building data structure becomes very simple. Just read the nodes and links in every mode network and inserting them into a list in computer memory by order. The users can determine the weights of various links with user interface before the program reads link impedance from the network to the memory. The final impedance values are equal to the value of Travel Time or Cost multiplied by the weight. For example, users want to walk on less distance, reduce the time of getting on and off and reduce the number of transfers due to his heavy luggage. He can set up a higher weight for the link of road network and dummy link. Thus the higher values could be gotten through the value that Travel Time multiplied by weight inputted. Then user will get a result path in which there are shorter walking distance and fewer transfers.

### 3.3 Work with the Constraints

Although each mode of network is assigned a weight before calculation, user's constraints could not be always satisfied. As the calculation processing some constraints will exceed its maximum. The weight of each network needs to be readjusted. Figure 3 illustrates the principle of adjusting weights. Path 1 and Path 2 are two paths in intermodal network from origin to destination. These two paths are composed of two modes: mode 0 and mode 1. Path

1 is shorter than Path 2 but the path of mode 0 in Path 1 is larger than that of Path 2. When a bigger weight is assigned to mode 0, the length of Path 1 and Path 2 will increase. Since the length of mode 0 in Path 1 increases faster than Path 2, the total length of Path 1 will finally exceed Path 2. Path 2 becomes the shorter one after adjusting the weight of mode 0. According to this principle, suppose user set a constraint of a maximum walking time. When this constraint exceeds the maximum during the k-shortest path generation, a bigger weight can be assigned to mode 0 until there is another path which can satisfy the constraint shorter than current one and that path will be selected as the k<sup>th</sup> shortest path instead. In practice the access links and egress links are assigned to a new mode for easily adjusting their weight.

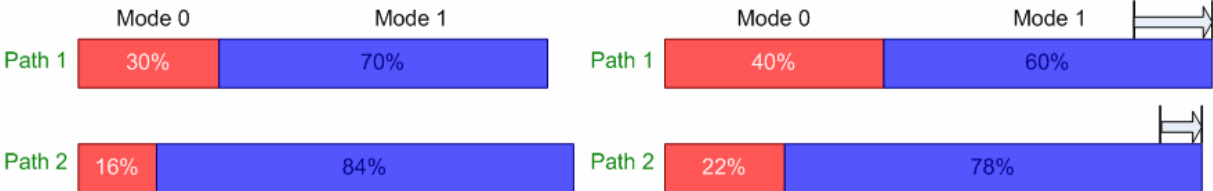


Figure 3 Two Paths before Adjusting Weights (Left) and after Adjusting Weights (Right)

### 3.3 The Improved Link Penalty Method

The core of the user preferable k-shortest path is the link penalty method. In general, link penalty method updates all of the links' impedance in a shortest path and recalculates a shortest path from origin to destination. This method is always used to generate dissimilar k-shortest path. But this study improved the original link penalty method to meet the user preferences. Figure 4 illustrate the outline of the improved link penalty method. A shortest path has been calculated from origin to destination first. Then calculate the length proportion of each mode in the whole path. In the case of figure 4, the mode 2 shares 47% of the total length and every link in mode 2 shares about 23.5% of the total length. That means mode 2 is the major mode in the three modes because its highest proportion in length. The link penalty will be applied on mode 2. A penalty value  $\delta$  will be added to the path of mode 2 and eventually every links in mode 2 will be added a penalty value. Perform a shortest path algorithm in this updated network and do the link penalty again. The generated paths maybe have groups of sub-paths in mode 0 and mode 1 and each group has the very high similarity among paths while in mode 2 paths have high dissimilarity.

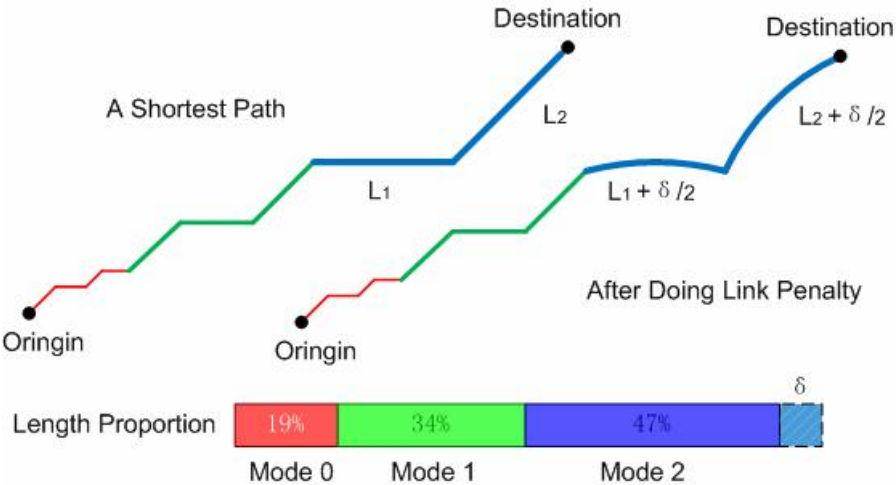


Figure 4 the Concept of the Improved Link Penalty Method

While figure 4 shows the basic principle of the link penalty method proposed by this study,

there are many issues posed. The most important issue is how to set the penalty value  $\delta$ . Suppose that the major mode contains  $n$  links, the length of these links cover from  $L_{\max}$  to  $L_{\min}$  evenly. The average interval of length  $I$  between two links is:

$$I = \frac{L_{\max} - L_{\min}}{n} \quad (7)$$

The shortest path and the 2<sup>nd</sup> shortest path in the major mode have at least one link of difference. That means the length difference between the current shortest path and the 2<sup>nd</sup> shortest path has a high possibility of being  $I$ . When set the penalty value  $\delta$  to  $I$  and recalculate a shortest path, the 2<sup>nd</sup> shortest path has the high possibility of containing one link not contained by the shortest path. Suppose there is a dissimilarity degree that has been expected on the 2<sup>nd</sup> shortest path. Thus:

$$d = \frac{\delta \times L_a}{I \times L_s} \quad (8)$$

, Where  $d$  denote the expected dissimilarity degree.  $L_a$  is the average link length in the major mode and  $L_s$  is the current shortest path in the major mode. Solve the equation 4 for  $\delta$ :

$$\delta = \frac{I \times L_s \times d}{L_a}$$

$$\delta = \frac{(L_{\max} - L_{\min}) \times L_s \times d}{L_{\max} + L_{\min}} \quad (9)$$

Since  $\delta$  is the penalty value of the whole path in major mode. Every link of the current shortest path in the major mode can get a penalty value of  $\delta/m$ , where  $m$  is the number of links of the major mode in current shortest path.

### 3.4 The Partial Overlap Criterion

Another big issue left by performing the improved link penalty method is the iteration criterion. Lim *et al.* (2005) described a path overlap method to calculate the proportion of the length overlapped by generated paths to the whole length of a path. In this study links will get link penalty only in the major mode network in each calculation loop. Paths in generated path set have dissimilarity in major mode and should have high similarity in minor modes. It is meaningless to perform an overlap calculation in all of the modes. This study suggests a partial overlap method which only calculates the overlap degree in the major mode network. Particularly the paths of the major mode in the shortest path in every calculation loop will be stored in a path set. In the end of every calculation loop the partial overlap degree will be checked by calculate the proportion of the overlapped length in the path set to the whole length of current path. If the value of the partial overlap degree exceeds the maximum value, the loop of path generation will go to the end.

Chapter 3.1 mentioned  $D(M(P(o,d)))$ , the dissimilarity indicator of the major link in link set, the return value of this function will be partial overlap. Let's start from the beginning. First we have a set  $a=\emptyset$ . Every end of iteration a path from  $o$  to  $d$  will be gotten  $P(o, d)$ . The major part of this path  $M(p(o, d))$  is a set of links  $b=\{L_1, L_2, L_3, \dots, L_n : n \in \mathbb{N}\}$ , the total length of links in  $b$  is

$$I(M(p(o, d))) = I(b) = \sum_n L_n \quad (10)$$

Those links then are compared with the set  $a$  to find which link has been overlapped. The overlapped links can be presented by  $c=a \cap b$ , the total length of links in  $c$  is  $I(c)$ , then the Partial Overlap is defined by

$$O = \frac{I(c)}{I(b)} \quad (11)$$

If the partial overlap is not exceed the criterion the major part of current path will be added to the set a, that is  $a = a \cup b$ .

### 3.5 The Formulation of the Algorithm

The data structure produced from the intermodal transportation network is just a bigraph; therefore, any standard Shortest Path Algorithm and K-Shortest Path Algorithm can be applied directly without any adjustment. The User Preferable K-Shortest Path Algorithm introduced by this study that uses improved link penalty method and partial overlap criterion have the advantages over other algorithms. It can identify the major mode of the intermodal network and calculate efficient k-shortest paths for user's preferences. First some variables are denoted as follows:

$P_{current}^{od}$	The current shortest path from origin to destination
$P_{major}^{od}$	The part of current shortest path in the major mode
$P_{set}^{od}$	The path set which can stores paths from origin to destination
$L^{od}$	The length of current shortest path
$L_{major}^{od}$	The part of $L^{od}$ in the major mode
$L_{major}^{overlap}$	The overlapped part of $L^{od}$ in the major mode
$L_{major}^{max}$	The maximum length of link in major mode network
$L_{major}^{min}$	The minimum length of link in major mode network
$L_{link}$	The length of a link
$N_{major}^{od}$	The number of links in $P_{major}^{od}$
$O_{major}$	The partial overlap degree in major mode
$O_{max}$	The maximum partial overlap degree
$S_l$	The proportion of a link's length in $P_{current}^{od}$ to $L^{od}$ , $S_l = \frac{L_{link}}{L^{od}}$
$W_{mode}$	The weight of impedances in a mode of network
$W_{mode}^{max}$	The maximum weight of impedances
$E_d$	The expected dissimilar factor
$\delta$	The link penalty value for the path in major mode

Then the procedure of calculation can be described as follows:

[Step 0]: Set  $O_{max}$ ,  $W_{mode}^{max}$  and  $E_d$

[Step 1]: Calculate the shortest path  $P_{current}^{od}$  from o to d

[Step 2]: If current path exceed one of the constraints and find  $W_{mode}$

If  $W_{mode} > W_{mode}^{max}$  go to step 6,

Otherwise adjust  $W_{mode}$  then go to step 1

[Step 3]: Calculate  $S_l$  for each link in  $P_{current}^{od}$  ;

Set the mode of the maximum  $S_l$  as major mode



[Step 4]:  $O_{major} = \frac{L_{major}^{overlap}}{L_{major}^{od}}$  ;

If  $O_{major} > O_{max}$  go to step 6 otherwise add  $P_{current}^{od}$  to  $P_{set}^{od}$

[Step 5]:  $\delta = \frac{(L_{major}^{max} - L_{major}^{mix}) \times L_{major}^{od} \times E_d}{L_{major}^{max} + L_{major}^{mix}}$  ;

Set  $L_{link} = L_{link} + \frac{\delta}{N_{major}^{od}}$  for each link in  $P_{major}^{od}$  ; Go to Step 1

[Step 6]: Output  $P_{set}^{od}$

This procedure will calculate k-shortest paths in the major mode of the intermodal network for single objective or a combined objective. The impedance of every link can be set to the equivalent value of the objective. If multiple objectives, such as time and cost, need to be reflected in results, one should be changed to a constraint and then perform this procedure. This procedure could be treated as a subroutine if the minor mode network also needs to be calculated hierarchically.

The core of the procedure is selecting the links to be updated and calculate the partial overlap. From users' point of view, when they want to find out some efficient travel plan in intermodal transportation network, they prefer to get some paths that can meet their preferences. To realize this purpose this study escalates a major mode network in intermodal network to maximize the user's preferences.

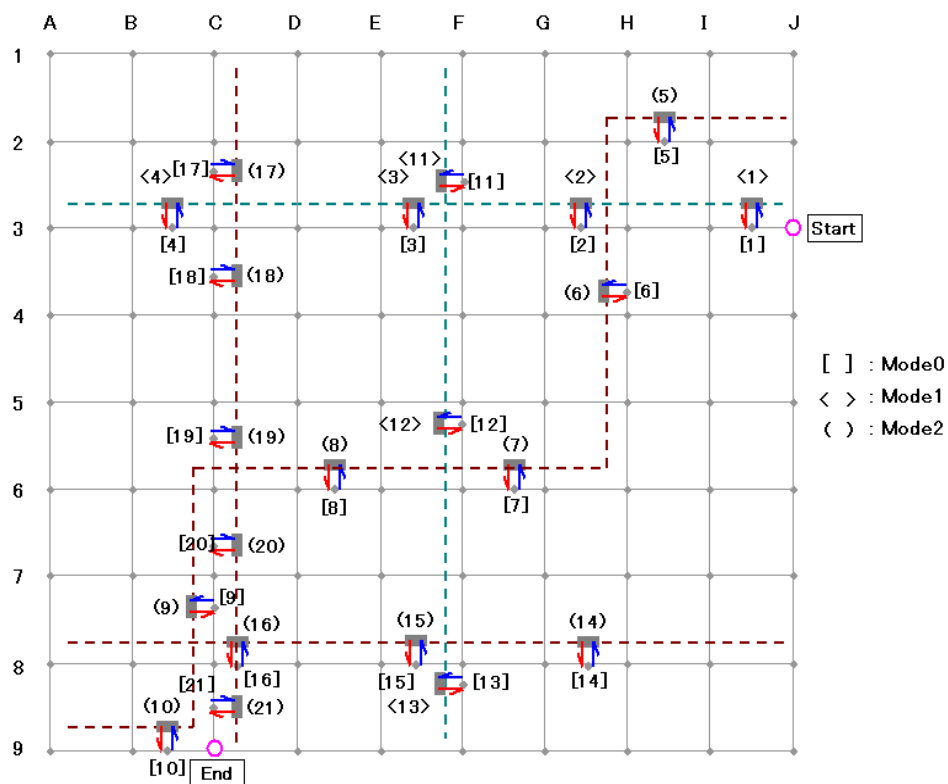


Figure 5 the Example of an Intermodal Transportation Network

If the transportation cost function is flow dependent then the impedances should be reread before every calculation. And if this algorithm is used in traffic assignment the impedances

should also be updated afterwards. If other variables are dynamic too it will not be so complicated, because we use an iterative algorithm. Every iteration could be treated as a static problem.

## 4. A NUMERICAL EXAMPLE

### 4.1 Build an Intermodal Transportation Network

For better understanding the algorithm proposed by this study a numerical example will be given in this chapter. First a road network has been created (Figure 5). For convenience the road network is created as a grid. Then another two modes of networks have been built separately. Note that the blue and red dotted lines are not along the road network. They can be treated as rail lines. A small black rectangle is used present the stops in these networks. The projection method is used to connect these three networks. Every stop and projected node has been assigned a number. The different decoration styles of the number represent different modes.

### 4.2 Network Initialization

Before perform the user preferable k-shortest path algorithm, the intermodal network has to be initialized. Table 1 lists all of the parameters used in the network and to be used in calculation. Note that each link is bidirectional and the impedance of each link can be treated as Travel Time.

Table 1 Initialization of the Numerical Example

Impedances	Mode 0	A1-A2 ... J8-J9: 10			A1-B1 ... I9-J9: 10		
		[1]-I3: 5	[1]-J3: 5	[2]-G3: 6	[2]-H3: 4	[3]-E3: 6	[3]-F3: 4
		[4]-B3: 5	[4]-C3: 5	[5]-H2: 4	[5]-I2: 6	[6]-H3: 6	[6]-H4: 4
		[7]-F6: 2	[7]-G6: 8	[8]-D6: 5	[8]-E6: 5	[9]-C7: 7	[9]-C8: 3
		[10]-B9: 3	[10]-C9: 7	[11]-F2: 8	[11]-F3: 2	[12]-F5: 8	[12]-F6: 2
		[13]-F8: 3	[13]-F9: 7	[14]-G8: 5	[14]-H8: 5	[15]-E8: 8	[15]-F8: 2
		[16]-C8: 3	[16]-D8: 7	[17]-C2: 6	[17]-C3: 4	[18]-C3: 5	[18]-C4: 5
	[19]-C5: 5	[19]-C6: 5	[20]-C6: 5	[20]-C7: 5	[21]-C8: 3	[21]-C9: 7	
	Mode 1	<1>-<2>:2	<2>-<3>:2	<3>-<4>:3	<11>-<12>:4	<12>-<13>:3	
	Mode 2	(5)-(6):7	(6)-(7):8	(7)-(8):6	(8)-(9):7	(9)-(10):5	(14)-(15):5
(15)-(16):5		(17)-(18):5	(18)-(19):5	(19)-(20):6	(20)-(21):5		
Access	5						
Egress	1						
Parameters	Constraints: Total Walking time <= 30min, Number of Transfers <= 2						
	Expected Dissimilarity = 0.5, Maximum Partial Overlap = 0.6						

### 4.3 Procedure and Results

To simplify the calculation procedure and make the results more clear, a table that lists the main calculation steps and results is given (Table 2 and Figure 6). For convenience and better understanding some minor steps will not be shown and the results will not be shown as scientific form.

Table 2 the Results in Every Step

The set of major links	{}
Calc the Shortest path	Start-[1]-<1>-<2>-<3>-<4>-[4]-[17]- (17) - (18) - (19) - (20) - (21) - [21]-End

Check Constraints	Walking time = 21 ≤ 30, Transfers = 1 ≤ 2
Find the Major Mode	Major Mode = Mode2 Links: (17)-(18)-(19)-(20)-(21)
Calc the Partial Overlap	Partial Overlap = 0 < 60%
The set of major links	{ (17)-(18)-(19)-(20)-(21) }
Perform link penalty	$\delta = 2.42$
Calc the 2 <sup>nd</sup> Shortest path	Start-[1]-<1>-<2>-[2]-[6]-<6>-<7>-<8>-<9>-<10>-[10]-End
Check Constraints	Walking time = 22 ≤ 30, Transfers = 1 ≤ 2
Find the Major Mode	Major Mode = Mode2 Links: (6)-(7)-(8)-(9)-(10)
Calc the Partial Overlap	Partial Overlap = 0 < 60%
The set of major links	{ (6)-(7)-(8)-(9)-(10), (17)-(18)-(19)-(20)-(21) }
Perform link penalty	$\delta = 3.00$
Calc the 3 <sup>rd</sup> Shortest path	Start-[1]-<1>-<2>-<3>-[3]-[11]-<11>-<12>-<13>-[13]-[15]-<15>-<16>-[16]-End
Check Constraints	Walking time = 29 ≤ 30, Transfers = 2 ≤ 2
Set the Major Mode	Major Mode = Mode2 Links: (15)-(16)
Calc the Partial Overlap	Partial Overlap = 0 < 60%
The set of major links	{ (6)-(7)-(8)-(9)-(10), (17)-(18)-(19)-(20)-(21), (15)-(16) }
Perform link penalty	$\delta = 0.58$
Calc the 4 <sup>th</sup> Shortest path	Start-[1]-<1>-<2>-<3>-<4>-[4]-[17]-<17>-<18>-<19>-<20>-<21>-[21]-End
Check Constraints	Walking time = 21 ≤ 30, Transfers = 1 ≤ 2
Find the Major Mode	Major Mode = Mode2 Links: (17)-(18)-(19)-(20)-(21)
Calc the Partial Overlap	Partial Overlap = 100% > 60% <b>End</b>

#### 4.4 Summaries

The results of this example contain four shortest paths. Only the first three are the user preferable paths. Mode 2 had been found as the major mode during the calculation. There are two constraints: walking time and transfers. They were always satisfied. Partial Overlap and Link Penalty worked well on controlling the process of the calculation.

## 5 CONCLUSIONS

In this paper a new k-shortest path algorithm named user preferable k-shortest path algorithm applied on intermodal transportation network has been introduced and tested. The user preferable k-shortest path algorithm can effectively evaluate the user's preferences and generate more humane paths for making a travel plan. The improved link penalty method and a partial overlap criterion compose the core of the algorithm and have been tested work well for meeting user's preferences and maximizing the user's objective. Because of the simple topological structure and the extendable k-shortest path algorithm, the developer can easily develop a rich featured traveler information system.

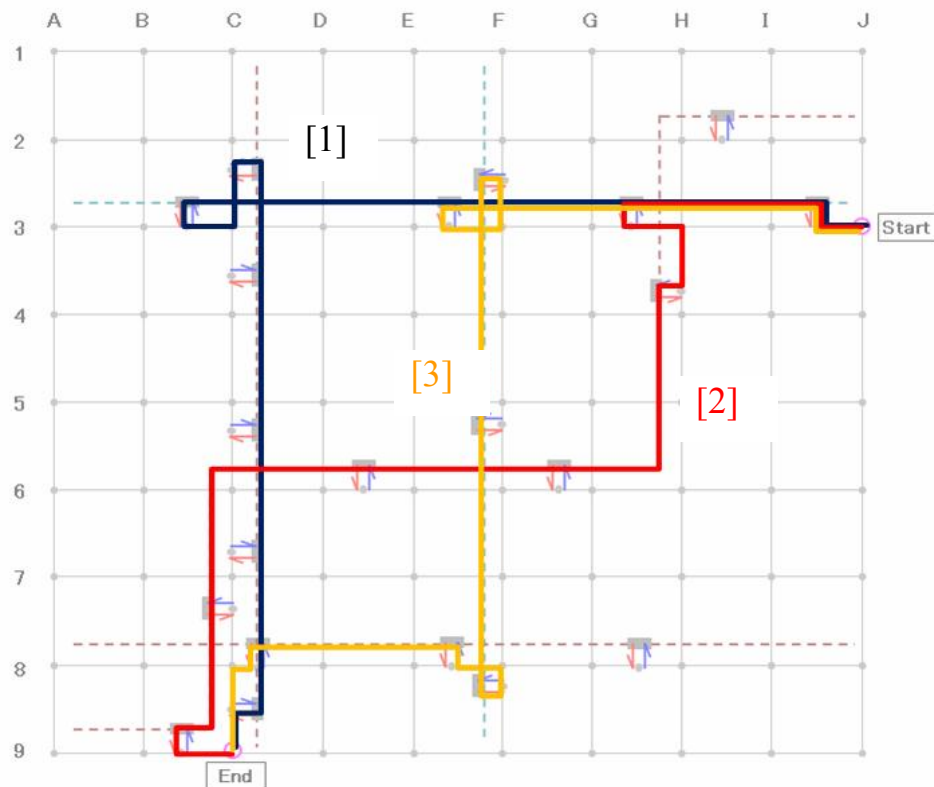


Figure 6 Three result paths (the fourth has been removed because of 100% partial overlap degree)

## REFERENCES

- Akgun, A., Erkut, E. and Batta,R. (2000). On finding dissimilar paths. **European Journal of Operational Research** **121**, 232-246.
- Athanasios, Z., Whitney, W. (2000). An Intermodal Optimum Path Algorithm for Multimodal Networks with Dynamic Arc Travel Times and Switching Delays. **European Journal of Operational Research** **125**, 486-502.
- Barra, T., Perez, B. and Anez, J. (1993) Multidimensional Path Search and Assignment, 21st PTRC Summer Annual Conference
- Liu, C., Pai, T., Chang, C. and Hsieh, C. (2001). Path-Planning Algorithms for Public Transportation Systems. *IEEE Intelligent Transportation Systems Conference Proceedings*. Oakland (CA) USA.
- Choi, K., Jang, W. (2000). Development of a transit network from a street map database with spatial analysis and dynamic segmentation. **Transportation Research Part C** **8**, 129-146.
- Mouskos, KC., Greenfeld, J. (1999). A GIS-Based Multi-modal Advanced Traveler Information System. **Computer-Aided Civil and Infrastructure Engineering** **14**, 267 - 279.
- Lin, L. (2002). Best-Path Planning for Public Transportation Systems. The IEEE 5th International Conference On Intelligent Transportation Systems 3-6. Singapore.
- Bielli, M., Boulmakoul, A. and Mouncif, H. (2006). Object modeling and path computation for multimodal travel systems. **European Journal of Operational Research** **175**, 1705 - 1730.

- Scott, K., Pabon-Jimenez, G. and Bernstein, D. (1997) Finding alternatives to the best path, Presented at the 76th Annual Meeting of the Transportation Research Board, Washington,DC
- Yen, J.Y. (1971) Finding the K shortest loopless paths in a network, **Management Science**, **Vol 17**, No. 11, 712-716
- Lim, Y., Kim, H. (2005). A SHORTEST PATH ALGORITHM FOR REAL ROAD NETWORK. **Journal of the Eastern Asia Society for Transportation Studies**, **Vol. 6** , 1426 - 1438.
- Sun, Z., Yang, Z.Z. (2005). A GIS-BASED PATH SEARCHING SYSTEM FOR TRANSIT NETWORKS. **Journal of the Eastern Asia Society for Transportation Studies**, **Vol. 6** , 2223 - 2233.
- Ziliaskopoulos, A., Wardell, W. (2000). An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays. **European Journal of Operational Research** **125** , 486-502.