Journal of Information Technology Management

ISSN #1042-1319

A Publication of the Association of Management

STUDY AND IMPLEMENTATION OF A DECISION SUPPORT SYSTEM FOR URBAN MASS TRANSIT SERVICE PLANNING

MAN-CHUN TAN
COLLEGE OF INFORMATION SCIENCE AND TECHNOLOGY, JINAN UNIVERSITY
GUANGZHOU, P R CHINA
tanmc@jnu.edu.cn

CONG-ON TONG
DEPARTMENT OF CIVIL ENGINEERING, THE UNIVERSITY OF HONG KONG
HONG KONG, P R CHINA

JIAN-MIN XU
COLLEGE OF TRAFFIC AND COMMUNICATION, SOUTH CHINA UNIVERSITY OF TECHNOLOGY
GUANGZHOU, P R CHINA

ABSTRACT

A decision support system (DSS) is developed for urban mass transit service planning, which is designed for several types of users, such as transit operators for service planning, a public agency for regulation of transit services, or transit passengers for route guidance. The DSS provides a flexible mechanism to deal with data. It may include various transit networks and one transit network has one corresponding database. An efficient algorithm for finding reasonable paths in a transit network is presented, which is used in the DSS for passenger's trip planning. The algorithm can be useful in various transit networks. A trip prediction model based on fuzzy neural network is employed in the system to forecast the number of trips entering or leaving a station. Several realistic transit networks in Hong Kong and Guangzhou are selected as case studies in different phases of the system development. To evaluate the performance of the system, some practitioners in transport planning and potential users were invited to participate in testing the system. The results of the prototype evaluation were satisfactory and the viability of the DSS as a useful tool for supporting decision has been ascertained by the positive feedback obtained from the survey questionnaires.

Keywords: decision support system; urban mass transit network; transit service planning

INTRODUCTION

Not only does the design and performance of a transit service system provide opportunities for mobility, but over a long term, it influences patterns of growth and the level of economic activity through the accessibility it provides to land. Transit service planning provides cooperation interaction among planning, design, and operation of transit services, and encourages alternative modes of transportation that will enhance energy efficiency while providing high levels of mobility and safety.

Guangzhou, capital of Guangdong province, is a modern metropolis in south China. It has more than 10.15 million people in 2003. More than 90% of its person-trips are by public transport, the potential benefit that can be gained from streamlining and coordinating various transit services is very great. There are lots of transit service providers in Guangzhou. Each of them can provide only the information of oneself, such as transit lines, schedules and fares. Travelers are not fully satisfied by the route guidance information provided presently by the transit service agencies. For example it is difficult for them to find where to transfer between different bus service providers. Therefore, a trip planning system is needed that integrates all information from all transit service providers.

In addition, competitive pressures make the job of decision making difficult. To achieve an effective transit system that maximizes community benefit, it is necessary to have a manager to regulate and coordinate transit services. To make effective decisions, this manager needs a tool to constantly track the performance of the transit system. A computerized Decision Support System (DSS) becomes necessary for the decision-makers.

In the paper, a DSS is presented for urban mass transit service planning. The proposed DSS is designed for several types of users, such as transit operators for service planning, a public agency for regulation of transit services, or transit passengers for route guidance. This prototype DSS involves several realistic representation of transit networks. In the first phase of the system development, the Mass Transit Railway (MTR) and Kowloon Canton Railway (KCR) networks in Hong Kong were selected as case studies. In the evaluation phase, a network including subways and bus lines in Guangzhou was selected. The uniqueness of the DSS lies in combining database management, trip planning, traffic simulation and analysis into a seamlessly integrated package. While neither of these elements is new, it is the tight integration that provides the new approach and qualifies the tool as a user-friendly decision support tool. An efficient recursive algorithm is put forward for finding reasonable paths in a transit network, which is used in the DSS for passengers to plan their trips. We believe the proposed DSS can not only help the transport manager or planner, help the Guangzhou tourism industry, but also the approach and methodology may be applied to overseas context.

The paper is organized as follows. In Section 2, we present a brief review of the literature review on the related area of the paper. In Section 3, the structure and function of the DSS is introduced. In section 4, the design and implementation of the data management sub-system is described. In section 5 and 6, two important models (trip planning model and trip prediction model) included in the model base are presented, where an efficient recursive algorithm for finding reasonable paths in a transit network is described in detail. In section 7, the DSS is employed in different scenarios and the analysis of evaluation is given. Section 8 concludes the paper and discusses further enhancements of the proposed DSS.

LITERATURE SURVEY

DSS research related to transportation

Transport system is so complex and dynamic that it is very difficult for transport planner to make decisions by traditional methods of dealing with data. In the 1970s, when it was recognized that management information systems did not address decision-making contexts very well, decision support system (DSS) technology was proposed (Turban [25]). DSS technologies have been developed to improve the effectiveness of managerial decision-making, especially in complex tasks (Bielli [3]). DSS are assisting transport planners in much many areas.

DSS could be a tool to evaluate the policies that would enhance the efficiency of transport supply with particular concern in improving environmental, social and budget indicators. Within a multi-modal transport system, the total travel demand must be balanced with the provision of reliable transport services. Model and DSS for planning urban transport policies have been put forward as a consequence (Arampatzis [1], Ferrari [8]). Colorni [6] proposed some practical applications of decision support systems for environmental impact assessment of transport infrastructures.

Logistics and transportation account for a large portion of the economics of the countries. A large number of papers have attempted to design a DSS for logistic decisions (Moynihan [19], Kengpol [16], Gayialis [10], etc.). Papers ranged from operational to strategic level, and from supply chain planning to Warehousing. About 70% of them supported lower-level tactical and operational decisions, and 30% represented strategic support systems.

Road safety and incident response are important subjects in the field of transport management (Benedetto [2]). Incident-related congestion is a serious problem of great concern for most metropolitan traffic management authorities. DSS for safety evaluation and emergency response operations have been proposed by a number of researchers (Zografos [28]).

DSS technology levels consist of specific DSS, DSS generators and DSS primary tools [25]. Most of the DSS mentioned above are specific DSSs, which are applications for some particular decision-making. Besides those in the specific DSS level, research in the DSS generator has also been proposed. Fierbinteanu [9] proposes a decision support systems generator for transport demand forecasting implemented by constraint logic programming, which supports the construction of a specific DSS from basic building blocks.

Model and method are the essential components of a DSS. The DSSs required to solve those transport planning problems rely not only on numerical analysis, but also on heuristic analysis in terms of words and concepts. New models and methods are continuously proposed to support transport planning, such as fuzzy expert system approach to handle uncertainty in the daily operation of urban bus transit systems (Chang [5]), and the approach of incorporating the time-of-use decision into a model of optimal pricing and service in urban mass transit (Kraus [17]).

Research on Trip planning and trip prediction

Route choice problem is a classic one in transit network (Hickman [12], Hall [11]). The classic shortest path problem consists of finding, in an oriented graph, a feasible path that links a given origin node to a given destination node and minimizes the sum of its arc costs. Much many articles on shortest path problem can be found in the literature. Tong [23] developed a schedule based transit network and proposed a time-dependent optimal path algorithm to finding the best path. Zhan and Noon [27] had produced a number of shortest path algorithms as well as some findings regarding the computational performance of the algorithms. Desaulniers and Villeaneuve [7] considered the shortest path problem with waiting costs as an extension to the shortest path problem with time windows.

A framework was presented for online transit information system and categorized the quality of service provided on different Web sites on the basis of content and functionality (Peng, et al. [21]). The content is divided into four levels (A, B, C, D) and the function levels range from 0 to 4, in which D4 is the highest level of content and function.

The optimal path is the one that minimizes the total travel cost, which is the weighted sum of several time components (such as waiting, riding and walking). Obviously this model mentioned above can be used for deterministic transit assignment and generating route guidance information for passengers. However, the route guidance information provided presently by many transit service agency are not really satisfying passengers. In trip planning, passengers don't always want to know which is the exactly best path between an OD pair, because the weights of waiting, walking time or fare may be different from each other and different for the same person on different purpose of the trip. What they want to know may be: how many reasonable (or feasible) paths there are between the OD pair, how much the trip time or fare is in each path, etc.

Ingmar Andeason [13] described a method for path finding in a headway-based transit network, in which the finding of one-transfer path is introduced. However Little effort has been spent on the reasonable paths finding.

Predicting the number of trips entering or leaving each station in a transit system is important in designing and controlling traffic facilities. Previous researchers on prediction have mainly employed linear statistical models such as AR model, the MA model, the ARIMA model, the Kalman Filtering model, and recently neural network models. Recently many researchers have claimed that neural models are much superior to traditional statistical models in forecasting future events (Yin [26]).

Neural networks, with their superb learning capabilities, play an important role in numerical processing and pattern recognition. The neural network approach has received extensive application in traffic engineering, a large number of researches can be found in the literature (Bielli [4], Shmueli [22]).

However, an ANN generally lacks the ability to be developed for a given task with a reasonable time, and it is difficult to come up with reasonable interpretation of the overall structure of network. On the other hand, fuzzy logic can provide an interesting and powerful scheme for knowledge representation, and requires an approach to learn from experience. In order to benefit from the strengths of each technique we could employ them both in combination rather than exclusively (Palacharla [20]).

Literature survey above gains a broad view of the kind of subject areas related to DSS, trip planning and prediction. Obviously the categories given above are purely arbitrary; some could reasonably be merged, whilst others cover what is arguably a very broad area and could be sub-divided.

Although there are plenty of researches in the DSS related to transportation, in the literature little effort has been found dealing with the computer-based mass transit service planning. An efficient recursive algorithm is proposed for finding reasonable paths in a transit network. The proposed DSS uses the algorithm for passengers to plan their trips. Although the trip planning application is only realized on a personal computer now, it will be a D4 level one (based on Peng [21]), if it is developed and deployed on the Web.

COMPONENTS AND FUNCTIONS OF THE DSS FOR TRANSIT SERVICE PLANNING

Components of the DSS

The proposed DSS for transit service planning is mainly composed of the following subsystems:

(1) Data management. It includes the database(s), which contains relevant data (station, line, timetable, fare, etc.) for transit system and is managed by software called database management system.

Information provided by public agencies may have different format. The database management system integrates all information from different service providers.

As a system for transit service planning, it may be used as a tool to evaluate the performance of the present network or a tool to design a new network in the future. As a result, the DSS may include various transit networks, such as networks in different cities, or networks at different time period in a city.

Although the networks are different from each other, their structures are really the same. Each transit network is stored in one unique database. That is, each network has a corresponding database. User can select one target network, before any of application in the DSS is launched.

(2) Model management. It is a software package that includes all models and an appropriate software management system. User can access and retrieve existing models and construct new models.

Now the models in the MBMS include trip planning model and trip prediction model, which will be described in detail in section 5 and 6. The relationship between the models and their data sources is shown in Table 1.

Table 1. Relationship between model and its data source

Model	**Data source (tables in the database)**
Trip planning	System, Station, etc. (shown in Fig.3)
Trip prediction	OD trip tables (shown in Fig.4)

(3) Dialog management. The dialog generation and management system (DGMS) provides a friendly user interface that allows the traveler or transit operator to query the database to obtain useful information to support their decisions. In the DSS, the combination of many interface modes has been adopted, such as: pull-down menu, question and answers, form interaction, graphics, etc.

Functions of the DSS

(1) Data storage, retrieval and update

- It should provide a logical structure for storing all relevant data.
- It should define the relationship between different types of data. Referential integrity is enforced in the database. It is a system of rules used by Microsoft Access to ensure that relationships between records in related tables are valid, and that user do not accidentally delete or change related data.
- It should facilitate data updates and data access. For instance, a dialogue for the update of Station table is shown in Figure 1.

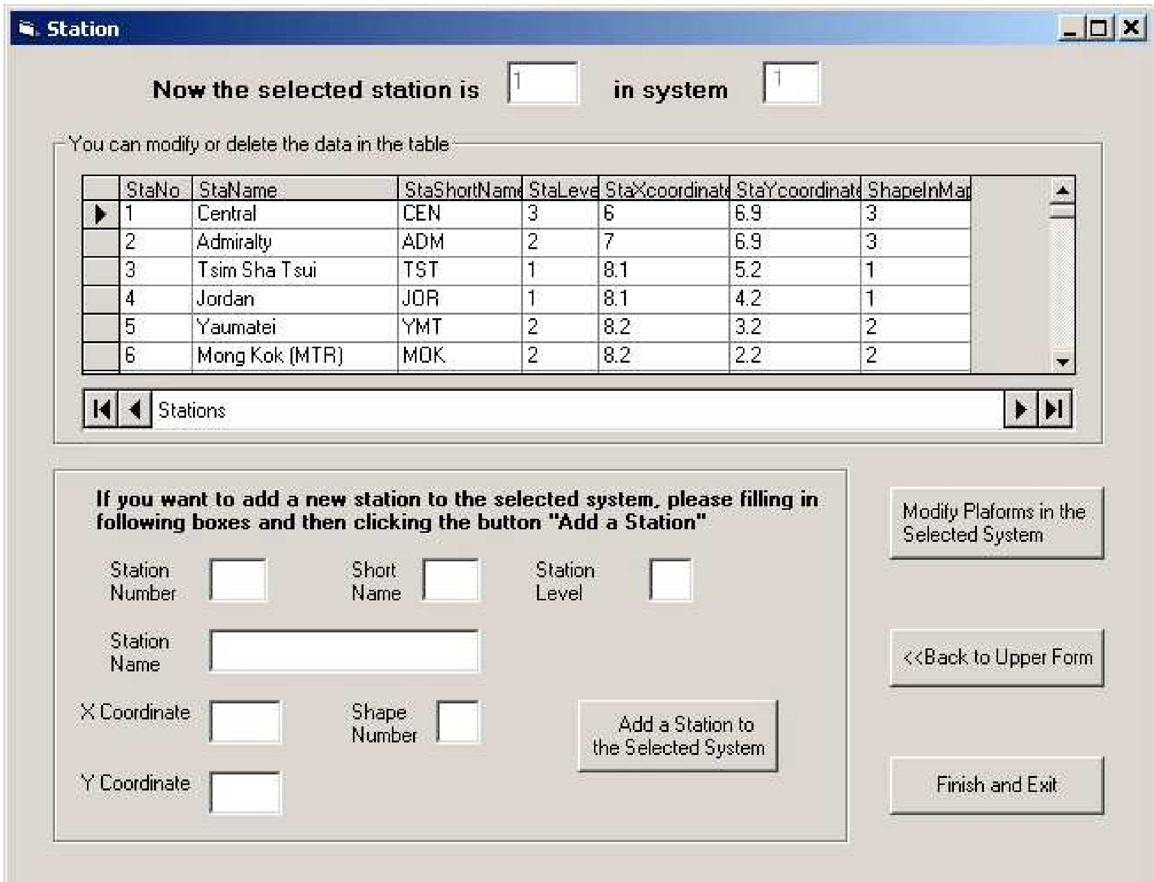

Figure1: Dialogue for the update of Station table

(2) Security manager

The security manager has the task of limiting access, modification, and malicious intrusion to the database. To perform these control actions the security manager requires that users be identified, authenticated, and authorized for the access and control over a data item being requested.

(3) Enquiry and Reports

Some information can be retrieved directly from the database, and some can be got by the applications designed for different enquiries. For example: the following data can be retrieved directly using normal database management system, such as Microsoft Access.

- First train and last train in each station according to the schedule;
- Fares (including all classes: adult, concessionary, etc.) between two stations in one system;
- All Lines in one system;
- All lines through a station;
- Stations in a transit line;
- Platforms in a transit line.

The following data can be got by some designed applications (or programs):

- The line-changing stations and walking time between different systems;
- Map (Diagram) of the transit systems;
- Minimum traveling time between any two stations in one system;
- Vehicle-minutes in one system;
- Shortest, cheapest path between two stations in different systems;
- Trip planning.

(4) Analysis and decision-making support

Models in the model base can be employed to analyze the performance of the transit service network and support decisions for planner and user.

IMPLEMENTATION OF DATABASE MANAGEMENT SUBSYSTEM

Database model

The first task of establishing the DBMS is the design of the database. There are several widely used DBMSs, such as ORACLE, Microsoft Access, INFORMIX, Sybase SQL, Visual Dbase 5.5, etc. Every DBMS supports one kind of data model (James [14]). Considering the following two factors, we select the relational data model as the database model in the conceptual design.

Firstly, compared with other models (such as network and hierarchical model), the relational model has a rigorous mathematical base and relational DBMS always provides an easy used SQL, which has a systematic, nonprocedural interface and removes the need for user to understand a host programming language. Although the transit system can be represented by a network or hierarchical model, the relationship of major entities (such as station, line, fare, etc.) can also be realized by a relational database.

Secondly, the relational model is compatible with existing application software packages used for transit service planning. Data in a relational database can be directly accessed by many existing applications.

The relational database and its normalization

In the first phase of this research project, the mass transit rail and the sub-urban rail systems in Hong Kong are selected as a case study for the database design. In the relational DBS of transit service, some of the entities (such as system, line, station, fare, etc.) and their attributes (fields) are shown in Figure2 and Figure 3. Most of the fields' names in the tables are easy to understand. As described above the database is designed for different potential users, some of the attributes are added for special use.

The database is processed by the approach of normalization. Normalization aims to remove redundant information and to provide unique identification for individual records.

All the tables are normalized to third form. Provided that the name and short name of a system are not functional dependent, the tables in Figure 3 are in third normal form. The third normalization is concerned with the relationship between the non-key fields, which have all non-key fields non-transitively dependent on the primary key, i.e., non-key fields must be dependent on the primary key and they must also be mutually independent. Why no further normalization (4NF, 5NF or further) is applied? The answer is that relations in third normal form are sufficient for most practical database design problems.

SysNo	SysName	SysShortName
1	Mass Transit Railway	MTR
2	Kowloon Canton Railway	KCR

SysNo	**LineNo**	LineShortName	LineLongName
1	1	ILE	Island Line Eastbound
1	3	KTE	Kwun Tong Line Eastbound

SysNo	**StaNo**	StaName	StaShortName	StaLevel	StaXcoordinate	StaYcoordinate
1	1	Central	CEN	3	6	6.9
1	2	Admiralty	ADM	2	7	6.9

Figure 2. Sample of records from the selected tables of the database

Some remarks about the fields of tables above are given below:

SysNo: Number of System; StaNo: Number of Station.

Station.StaLevel: the sum of levels in a station, which can be seen in the input form of platforms. Most station have only one level. When a station is a line-change station in MTR or KCR, its level is mostly 2. In particular, the level of Central Station is 3.

Station.StaXcoordinate, StaYcoordinate: horizontal and vertical coordinates of a station in the map.

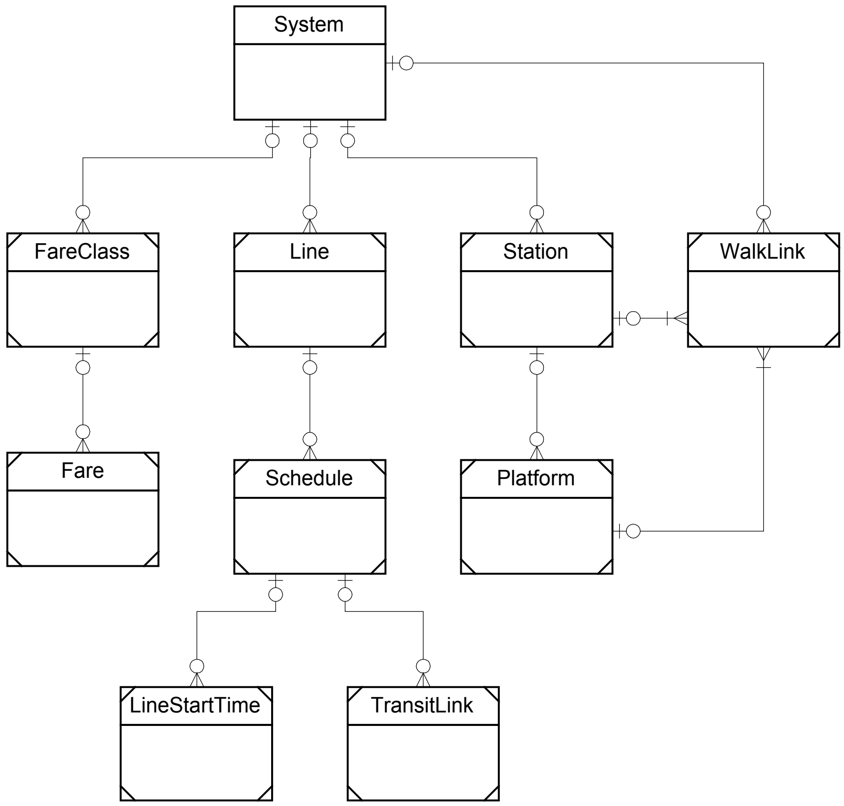

Figure 3: Database schema showing some tables and relationships among them.

Relationship diagram

The relationships among the entities in transit systems may be one-to-one, one-to-many, and many-to-many. The relational model requires that all data structures be represented by sets of tables (entities) that can be have only one-to-one or one-to-many relationships with each other. Therefore, each many-to-many relationship is converted to a pair of one-to-many relationships by creating an intersection entity class. The relationships among some tables in the DBS are shown in Figure 3.

OD trip tables

The OD trip tables in the database are all tables related to the number of trips between any Origin-Destination pair, which are data source of the trip prediction model. Now the OD trip tables consist of OD_minute, Trip_minute, OD_day, Trip_day, etc.

The approach of designing OD tables is based on the idea of designing a data warehouse. The OD trip tables are organized by time dimension. In a time dimension, a hierarchy might be used to aggregate data from the minute level to the day level, and to month level, etc. Within the time dimension, minutes rolls up to days, days rolls up to months, months rolls up to quarters, quarters roll up to years, and years roll up to all years, which is shown in Figure 4.

For example: In the database, OD_minute, OD_day and Trip_minute table has the following attributes respectively:

OD_minute:Date,timeInterval,fromSys,fromStation,toSys,toStation, numofTrips1.

OD_day: Date, timeInterval, fromSys, fromStation, toSys, toStation, numofTrips2.

Trip_minute: Date, timeInterval, SysNo, StaNo, In_trip, Out_trip.

OD_minute is a table where numbers of trips from Origin station (fromSys, fromSta) to Destination station (toSys, toSta) are stored. The trips can be collected every 5 minutes from the detector or counting in each station. OD_days stores trips each day. Trip_minute stores In_trip (numbers of trips entering one station) and Out_trip (numbers of trips leaving one station) in a time interval.

For any given corresponding condition (such as the same Date, timeInterval, fromStation and toStation, etc.), the following equations hold obviously:

numofTrips2 (number of trips in OD_day) = sum of numofTrips1 (number of trips in ODminutes).

In_trip (in Trip_minute table) = sum of numofTrips1 in ODminutes

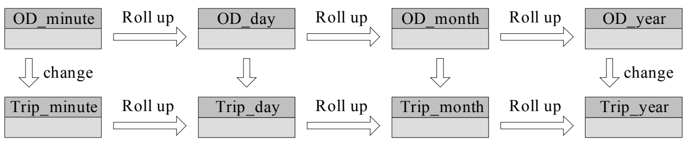

Figure 4: OD trip tables

TRIP PLANNING MODEL: FINDING REASONABLE PATHS IN TRANSIT SYSTEM

Introduction to trip planning models in the proposed DSS

There are two approaches (models) of trip planning in the proposed DSS. One is to find the best path for user. Tong [23] developed a time-dependent optimal path algorithm associated with a transit network. A path is considered to be optimal if it consumes the least weighted time, taking into account waiting time, walking time, in-vehicle time and line change penalty. The weights for the various trip components are defined relative to the in-vehicle time of a journey. The algorithm consists of three stages (Tong [24]):

Stage 1: Forward pass of quickest path;

Stage 2: Backward pass of quickest path;

Stage 3: A branch and bound method.

Sometimes passengers may want to know: how many reasonable (or feasible) paths there are between the OD pair, how much the trip time or fare is in each path, etc.

Therefore, the other approach is to find all reasonable paths. In each found path the in-vehicle time, walking time, waiting time, fare of each path will be provided to user, and the user can make his/her own decision by all the information.

In the following two questions are answered. Which path is a reasonable one and how to find it? Is the proposed algorithm efficient enough to find all reasonable paths?

where ((fromSys, fromStation) = (SysNo, StaNo)).

OD tables with different levels of granularity can be generated in advance, so user do not need to aggregate data each time from table to table when inquire the database.

Algorithm for finding reasonable paths in a transit network

Notation and Definition

A schedule-based transit network consists of a set of nodes *A*, a set of transit lines *L*, a set of walk links *W*, a pre-determined fixed schedule *FS*, and a time period of analysis *TP*. This transit network can be abstracted into a graph (*A, L, W, FS, TP*). A node is used to represent a source or sink of passengers, or to represent a transit stop, or a platform in a transit station.

For brevity, some new notations and definitions must be introduced below.

A walk link is regarded as a special line between two nodes, so transit line and walk link are called ***generalized line***.

G :set of generalized line, $G = L \bigcup W$.

$S(l)$: set of all nodes (or stops) in a transit line l . $S(l) = \{s_1, s_2, ..., s_n\}$.

$L(s)$: set of all transit lines via node s , $L(s) = \{l \mid l \in L \text{ and } s \in S(l)\}$.

Node r and s are in the same transit line if only if there exists at least such one transit line l that $r \in S(l)$ and $s \in S(l)$.

Node r and s is in the same generalized line if only if (1) there exists at least such one transit line l that $r \in S(l)$ and $s \in S(l)$, or (2) there exists a walk link between r and s .

A path is one that a user can follow on the transit network in order to travel between two nodes. ***A possible path*** between node r and s is one that doesn't pass any node more than one times.

A possible path from node r to s consists of a finite ordered sequence of nodes $< s_1, s_2, \cdots, s_n >$, where $r = s_1$ and $s = s_n$. For simplification and definition, the sequence $< s_2, \cdots, s_{n-1} >$ is called a possible ***via-node list*** if all nodes in $< s_1, s_2, \cdots, s_n >$ satisfy that (1) every pair of adjacent nodes is in the same generalized line and (2) every three adjacent nodes are not in the same transit line. The ***dimension*** of via-node list $< s_2, \cdots, s_{n-1} >$ is the number of nodes in it, so its dimension is *n-2*.

In a transit network, from one node to another there may exist many paths. Some of them perhaps need too much transfer in transit lines or need too long travel time. Therefore we only consider the reasonable paths. ***A reasonable path*** between a pair of OD nodes is any path that satisfies three conditions: (i) being a possible path, (ii) the dimension of via-node list in the path being no more than a given maximum number δ, (iii) $T - T_{\min} < \tau$, where

$T =$ Total trip time of the path,

$T_{\min} =$ The minimum trip time of all possible paths,

$\tau =$ certain given positive constant.

τ is also called an ***acceptable criterion***. When $T - T_{\min} < \tau$, user may consider the path to be acceptable.

δ is a positive integer given by the user. Because some walk links may be included in a via-node list, δ is often no less than the maximum transfer times.

For example, a simple transit network is shown in Figure 5. From node s_1 to s_4, there are several possible paths. Whether one of the possible paths is reasonable or not depends on δ and τ. All possible paths and their via-node lists are shown in Figure 6.

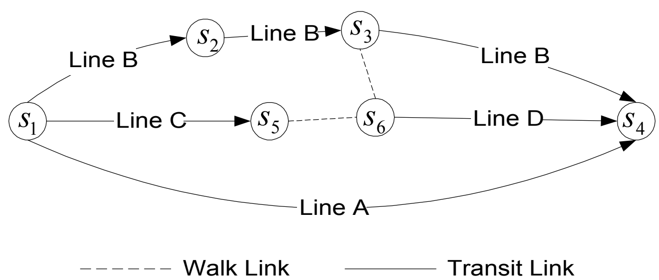

Figure 5. A sample of transit network

Possible Path	via-node list
s_1 —Line A→ s_4	EMPTY
s_1 —Line C→ s_5 - - - s_6 —Line D→ s_4	$< s_5, s_6 >$
s_1 —Line B→ s_2 —Line B→ s_3 —Line B→ s_4	EMPTY
s_1 —Line C→ s_5 - - - s_6 - - - s_3 —Line B→ s_4	$< s_5, s_6, s_3 >$
s_1 —Line B→ s_2 —Line B→ s_3 - - - s_6 —Line D→ s_4	$< s_3, s_6 >$

Figure 6. Possible path and its via-node list

$e(s)$ is called a ***set of transfer node*** $r (\in A)$, $r \neq s$, that satisfies one of the two conditions: (1) there is walk link between node r and s; OR (2) node r and s are in the same transit line, and there is at least one generalized line via node r but not via s. In other words, $e(s)$ is a set of transfer nodes where transfer may occur when passengers coming from node s.

Algorithm for Searching Reasonable paths

In common sense, every passenger hopes the transfer times in a path are as few as possible. Therefore, in the process of finding reasonable paths, the maximum transfer times is needed to be input or given before. In the case study its default value is 4.

The procedure is divided into five steps to find all reasonable paths between a pair of OD nodes if the starting time is given. A similar procedure can be used to deal with the problem if the arriving time to destination is specified. The major modules are (1) loading network, (2) searching reasonable paths and (3) reporting detail of each path.

Step 1. Load network.

All of the components (*A, L, W, FS, TP*) in a schedule-based transit network can be stored in a relational database, so the set $A, L, W, G, S(l)$ can be directly generated by reading the data in database. Since $S(l)$ is the set of all nodes in a transit line l, it is easy to generate all sets $L(s)$, set of all transit lines via node s. Then generate $e(s)$.

Therefore, in the module of loading network, its input is the data in database and output is $A, L, W, G, S(l)$, then $L(s)$, $e(s)$.

Step 2. Input some parameters. The user will be asked to provide the following information:

(i) origin and destination (OD) nodes;

(ii) the month, starting time from the origin;

(iii) δ and τ. The default values of δ or τ will be used by the application, if user doesn't provide or change them.

Step 3. Find the via-node list of each possible path. A recursive algorithm is developed to find the via-node lists of all possible paths. Its pseudocode is shown in Fig.7. colLineUsed and colStaUsed are temporary lists to store the generalized lines or stations, which are used to avoid a path passing through one line more than two times. PsPath(pathNo) store the detail information of a possible possible path numbered pathNo. The elements of PsPath are pDim, viaNodeList and GeneralizedLine. pDim and viaNodelist are the dimension and via-node list of a possible path, respectively. GeneralizedLine stores the generalized lines that are used in the path.

```
Algorithm FindViaNodeList
   Input: Origin Node, Destination Node, δ ;
   Output: Pspath(pathNo)
           { pDim as integer,   'dimension of via-node list
             viaNodeList as collection,
             generalizedLine as collection, 'lines used in the path
           }
Private sub FindViaNodeList(Sta1 As StationType, Sta2 As StationType,
                                   iDim As Integer)
  If iDim > δ Then
    Exit Sub
  End If
  For Each elem In e(Sta1)
    tpLineNum = LineBetween2Sta(Sta1, elem)
    If Not IsThreeStationsInALine(Sta1, elem, Sta2) AND
     Not (tpLineNum is in ColLineUsed)) Then
      ColStaUsed.Add elem     'add a station into collection
      If LineBetween2Sta(elem, Sta2)>0 Then
        RecordTheStationUsed iDim
      Else
        ColLineUsed.Add tpLineNum    'add tpLineNum into a set
        FindViaNodeList elem, Sta2, iDim + 1
        ColLineUsed.Remove (iDim)    'remove tpLineNum from the set
      End If
      ColStaUsed.Remove (iDim)  'remove a station from collection
    End If
  Next
End Sub
```

Figure 7: Algorithm for finding via-node list and its pseudocode

Step 4. Find generalized lines that pass through each pair of adjacent nodes in each possible path and calculate in the line the arriving time, waiting time, in-vehicle time, walking time, provided that each passenger will board the first arriving vehicle.

Suppose that node r_i and r_{i+1} are a pair of adjacent nodes. Because the starting time from the origin or the arriving time to destination of the user is known, the following data will be calculated between adjacent nodes (r_i, r_{i+1}):

(i) PsgWalkTime(i): Walk time between r_{i-1} and r_i;

(ii)PsgWaitTime(i): Time from the user arriving at r_i to vehicle departing;

(iii) InVehTime(i): In-vehicle time between node r_i and r_{i+1};

(iv) PsgArriveTime$(i+1)$: Time when arriving at station r_{i+1}.

Step 5. Select reasonable paths from possible paths and report the detail of each one including trip time, fare, etc.

Case study of trip planning

The MTR and KCR networks in Hong Kong are selected as the case study. The following example is finding all reasonable paths in the two transit networks.

Selecting “Find reasonable path” in the menu of the proposed DSS, a dialogue will be shown as Figure 8.

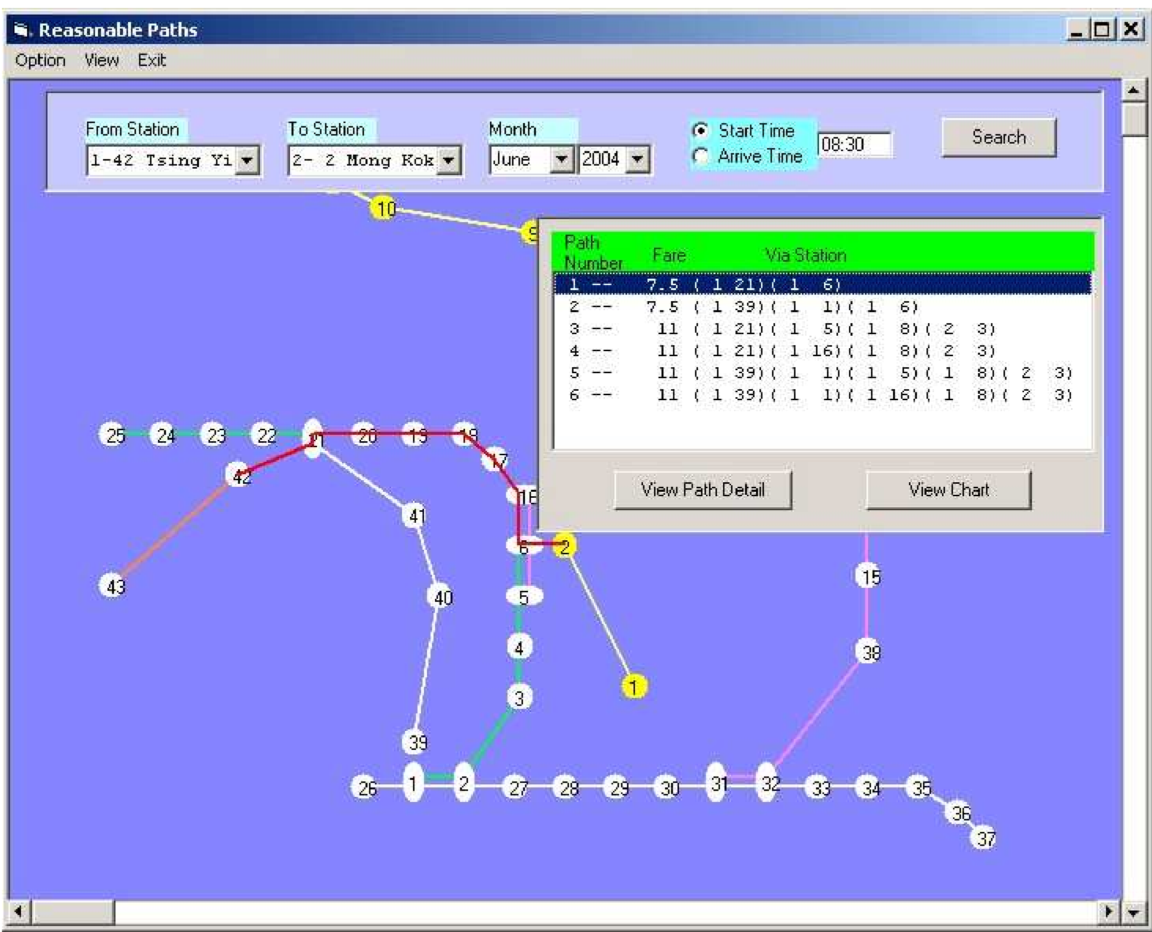

Figure 8: Dialogue for finding reasonable paths

On an option menu in Fig.8 by selecting "Change acceptable criterion", δ or τ can be changed. Obviously the bigger the acceptable criterion τ, more reasonable paths may be found out. Table 2 shows number of reasonable paths found on different setting of τ, where (1,42) is the (SysNo, StaNo) of station Lai King.

Supposed that origin is station (1,42) Lai King, destination is (2,2) Mong Kok, τ =30 and starting time is 8:30, then all reasonable paths can be found out by clicking button "search". Fig.9 shows the fares and via-node lists of all 6 reasonable paths in the left box, where the cheapest path can be easily found out. By clicking the button "View path detail", the itinerary can be shown in the right box.

Table 2: Example of τ and paths (start time 08:30 am)

Origin (From-Station)	Destination (To-Station)	Acceptable Criterion τ (min)	Number of Reasonable Paths found
(1,42) Lai King	(2,2) Mong Kok	30	6 (see figure 9)
		10	3
(1,3) Tsim Sha Tsui	(2,5) Sha Tin	30	8
		10	4

By pressing the button "View Chart" in Figure 9, a chart of details of all reasonable paths will be shown as Figure 10, which can be used to compare the all paths with each other, such as fare, total trip time, etc.

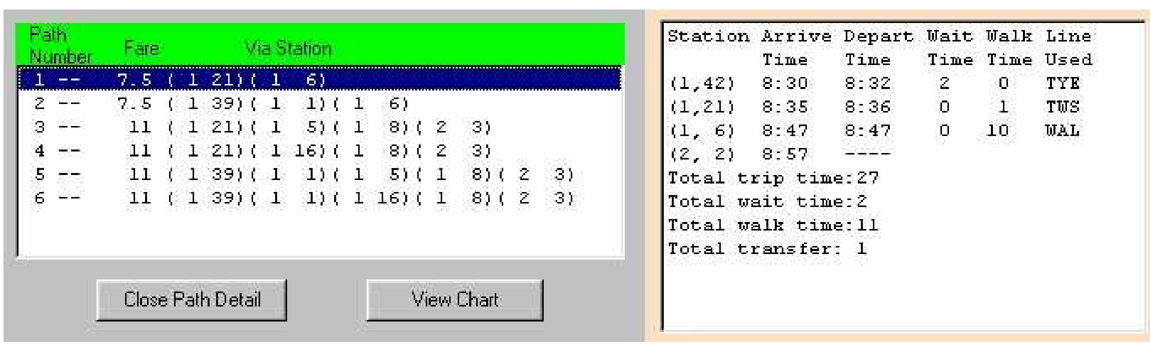

Figure 9: Reasonable path and its detail

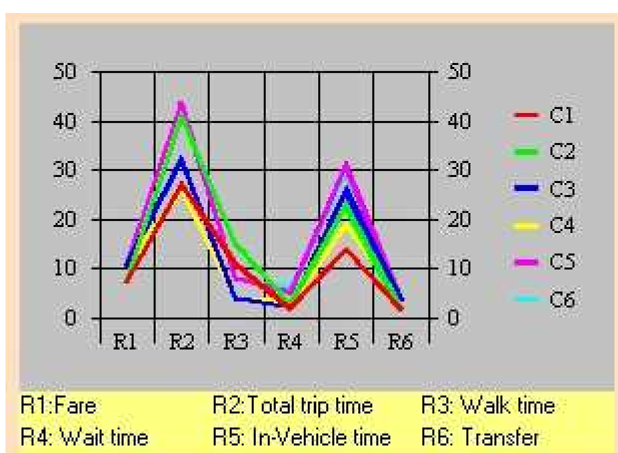

Figure 10: Comparison of all reasonable paths

The in-vehicle time, walking time, waiting time and fare of each found path have been provided to user, and the user can make his/her own decision by all the information. User can easily find out which path is quickest or cheapest from Fig. 8 to Figure 10.

The case studies show that the algorithm is efficient enough. In a Pentium III personal computer, the average time of finding reasonable paths between any OD pairs is no more than 0.5 seconds, just immediately after the button 'Search' in Figure 8 is clicked.

PREDICTION MODEL: TRIPS PREDICTION BASED ON A FUZZY-NEURAL NETWORK

Introduction to prediction based on FNN

In the transport planning context, there is a need to assess the total level of demand at a network. It is important to predict the number of trip entering or leaving a station in a transit system. Fuzzy neural network methodology provides a tool through which to model intuitive judgments without the complication of having to formalize all of the complex causal variables and relationships which other models require.

In the proposed DSS a fuzzy neural network (FNN) model is employed to predict the number of trips in a transit network. The trips on a link can be predicted from historical data or the real-time data collected from the detectors installed inside the stations in the past interval. Only the prediction of trips entering a station is described here, since the approach is completely similar to the predicting of that leaving a station.

Let $Q(i,t)$ is the trips entering a station i within time interval t, and

$$Q(i,t) = F[Q(i,t-1), Q(i,t-2), \cdots, Q(i,t-n)] \quad (*)$$

where n is a positive integer. The function $F(\cdot)$ specifies the generally nonlinear relationship between the historical data and the output data.

To model the highly nonlinear function $F(\cdot)$, a multilayer NN could be used to specify the input-output relationship. However, the training time of such a NN may be too long, since the traffic patterns are much different, and the parameters may change back and forth to achieve the best matching between modeled and target results. To reduce the training time, the traffic situations could be classified into several classes in advance, and a particular NN is dedicated to each class (Ledoux [18]).

In the proposed DSS the traffic situations (here the trips entering a station) are classified into a set of fuzzy cluster using fuzzy approach. The fuzzy neural network generates the fuzzy rules from historical data while learning. The adaptive rules are capable of approximating any trip prediction to good accuracy.

The fuzzy-neural model

The structure of the employed FNN is shown in Figure 11. There are n input nodes labeled $x_1, x_2, \cdots, x_n$ and one output neuron called y.

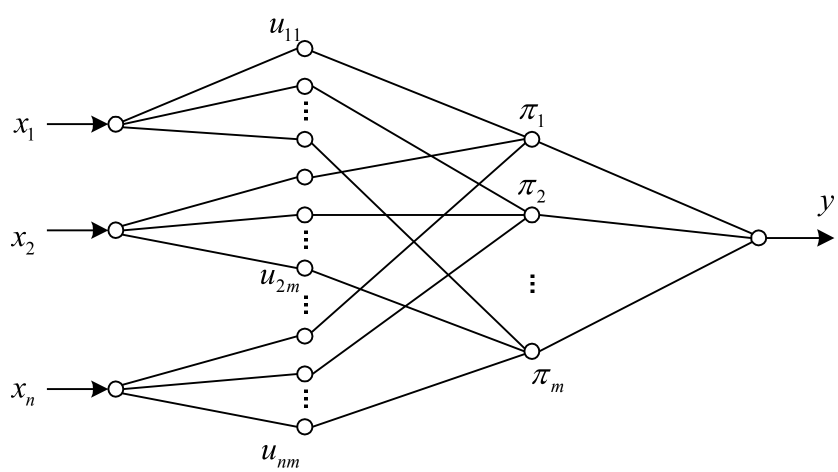

Figure11: Structure of the FNN model

Each input component in the input set $x_i(i=1,2,\cdots,n)$ is fuzzified into *m* components using the following equation based on the Gaussian base:

$$u_{ij}(x_i)=\exp\left[\frac{-(x_i-m_{ij})^2}{\delta_{ij}^2}\right],$$

$$1\le i\le n,\quad 1\le j\le m,$$

where m_{ij} and δ_{ij} are used to modify or 'tune' the membership function to be defined later, *j* corresponds to the *j*th rule (*j*th hidden node). The individual rule results are given by

$$\pi_i=u_{1i}\cdot u_{2i}\cdot\cdots\cdot u_{ni}=\prod_{j=1}^{n}u_{ji}\ ,\ 1\le i\le m\,.$$

The weights for the FNN are denoted by $w_i(i=1,2,\cdots,m)$, and the output of the FNN is given by

$$y=\sum_{i=1}^{m}w_i\pi_i\ .$$

The error function *E* is defined as

$$E=\frac{1}{2}(y-Y)^2$$

where *y* is the predicted output of the network, and *Y* is actual number of trips.

The back-propagation algorithm is used to train the FNN. The membership functions and the weights are tuned during learning using the following equations:

$$\Delta m_{ij}=-\eta\frac{\partial E}{\partial m_{ij}}$$

$$=-2\eta(y-Y)w_j\cdot\frac{x_i-m_{ij}}{\delta_{ij}^2}\cdot\exp\left[\frac{-(x_i-m_{ij})^2}{\delta_{ij}^2}\right]\cdot\prod_{l=1,l\ne i}^{n}u_{lj}$$

$$\Delta\delta_{ij}=-\eta\frac{\partial E}{\partial\delta_{ij}}$$

$$=-2\eta(y-Y)w_j\cdot\frac{(x_i-m_{ij})^2}{\delta_{ij}^3}\cdot\exp\left[\frac{-(x_i-m_{ij})^2}{\delta_{ij}^2}\right]\cdot\prod_{l=1,l\ne i}^{n}u_{lj}$$

$$\Delta w_{ij}=-\eta(y-Y)\pi_i$$

Finally, the procedure to the trip prediction is summarized in the following:

Step 1. The database is divided into five different sets corresponding to the five trip patterns to be modeled: Very Small, Small, Middle, Large, and Very Large.

Step 2. The traffic situations (here the trips entering a station) are classified into a set of fuzzy cluster using fuzzy approach. For a given forecast time-interval, the values for the input nodes are identified. These values are then compared with historical values for the same trip pattern using a selection procedure wherein the values are compared with historical values and the potential training cases are then selected.

Step 3. The network is trained until the observed error is minimized.

Step 4. The input values for the forecast time-interval are applied to the network and a prediction is produced.

Case study of trip prediction

In MTR in Hong Kong, trips entering or leaving a station can be collected every 5 minutes by the ticketing barriers in stations.

After the accuracy of results is analyzed based on different *n* (number of inputs) and *m* (number of fuzzy sets connected to each input node), it is observed that *n*=3 and *m*=5 is appropriate for the problem.

In the proposed DSS a step-by-step procedure is given to predict the number of trips entering or leaving a station by a FNN model.

(1) Select historical data.

User can predict the trips in a particular time-interval, a day, one month or one year, by selecting historical data of different types in the database. In the database the trips in every 5 minutes were collected in a table, and trips in every day, or every month were also calculated and stored in other tables.

For example, a user selects the data from a table that stores trips entering in a station every 5 minutes, the observed trips and the trips predicted by the FNN (240 minutes, from 6 am to 10 am) are shown in Figure12.

(2) Load past training result. If no new data is input into the database, the past training result can be loaded and used again.

(3) Re-train the FNN. If the database is updated, the user has to re-train the FNN.

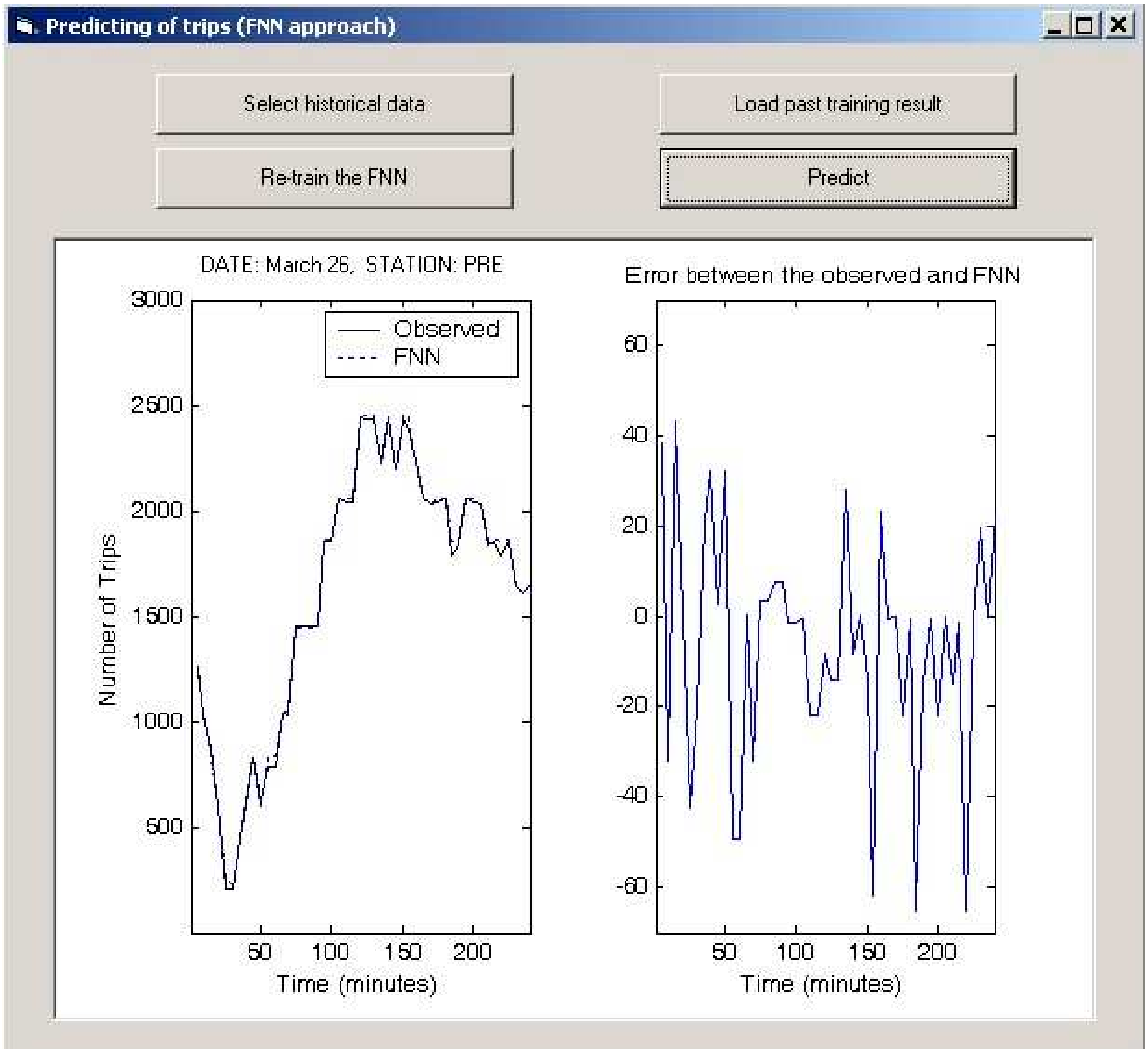

Figure 12: The observed and predicted results in 240 minutes.

Discussion on FNN

The proposed FNN has many advantages over a conventional fuzzy system. A conventional fuzzy reasoning system, which has *n* inputs and where each domain is partitioned into *p* fuzzy sets, requires p^n rules. The proposed FNN architecture requires only *pn* rules.

In equation (*), $Q(i,t)$ is the nonlinear function of the historical data within past time interval $t-1, t-2$, …, $t-n$. After the comparison of the accuracy of prediction based on different *n* was carried out, it is observed that the bigger *n* doesn't result in a more accurate prediction, and *n*=3 is appropriate for the simulation.

In general the prediction capability (accuracy) of a fuzzy reasoning system is proportional to its granularity (the number of fuzzy sets). However, the analysis indicated that there was no significant increase in the prediction accuracy when increasing the granularity above 5 fuzzy sets on each input. Furthermore increasing *m* not only increases the network dimensions and training times but it may also over-parameterize the problem, which may degrade its performance.

The training of FNN was quick enough to predict on-line. Once the data in the past intervals were collected, the trips can be predicted (including re-training the FNN) within one second in a Pentium III PC. The sensitivity analysis of the fuzzy neural model will be presented in detail in a future paper.

DSS IN DIFFERENT SCENARIO AND ITS EVALUATION

DSS in different scenario

As mentioned before the proposed DSS may include many networks in different time period, such as network at present, 5 years later or 10 years later. The traffic network is organized along time dimension. In a specific time each city has one network. Each transit network is stored in one unique database. When launching the DSS, user can select a database or use the default database without changing anything else. The main menu and the dialogue for selecting a database is shown in Fig.13.

The testing and evaluation of the proposed DSS mainly includes two phases of evaluation: (1) prototype testing and system evaluation and (2) outcome evaluation. Different scenarios were used in the two phases:

Scenario 1: a network including the Mass Transit Railway (MTR) and Kowloon Canton Railway (KCR) networks in Hong Kong.

Scenario 2: A network including 2 subways and 23 bus lines in Guangzhou.

In the first phase, two systems (the MTR and KCR network in Hong Kong) were selected as the case studies for the database design of the proposed prototype DSS.

In the evaluation phase, a network (including two subway lines and 23 bus lines) in the urban zone of Guangzhou was selected as case study to evaluate its effectiveness and usability. The two subway lines managed by Guangzhou Metro Corporation have been in operation for more than two years. 23 Bus lines were selected from those operated by Guangzhou No. 2 Bus Company. It is a bus transport backbone enterprise and launches over 170 lines, which cover the main cities and towns in Pearl River Delta in China.

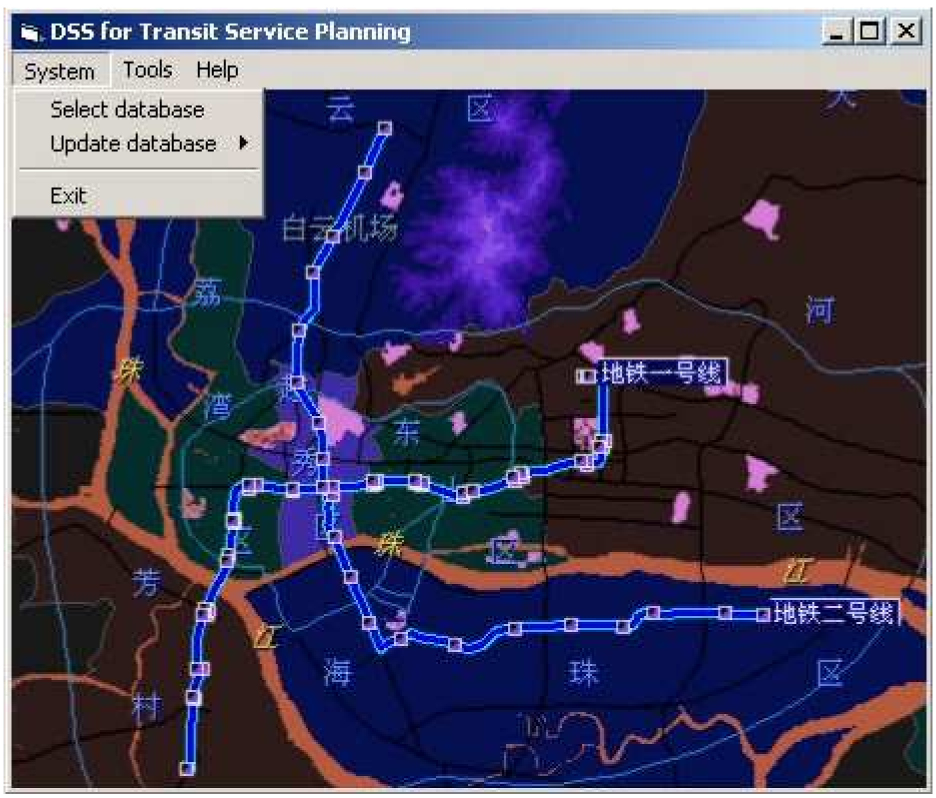

Figure 13. The main menu and a dialogue for selecting a database.

System evaluation and prototype testing

In the phase of prototype testing and system evaluation, all the modules of the DSS were tested for accuracy and completeness by the development team during system development. Prototype testing concerns an evaluation of the system by domain experts and non-implementers. The evaluation focuses on the impressions and attitude toward the design features and capabilities of the proposed system. In 2003 two academicians were invited to evaluate it. Later, 40 students, who major in the Information Management and Information System in the College of Information Science and Technology at Jinan University, were invited to participate in the evaluation.

The objective of outcome evaluation is to assess the overall value of the DSS. Outcome evaluation consists of two phases: potential users' evaluation and planner's evaluation.

Potential users (Tourists) focused on the evaluation of the trip planning sub-system (including trip planning module and its database). We randomly selected a total of 80 potential users, coming from 12 provinces of China, who had just arrived at the Guangzhou Railway Station during the period of 20-25 June 2004, and they are agreed to participate in the evaluation.

Transport planner's evaluation was conducted for the whole system. It was evaluated by two academics and 23 part-time Master degree students who major in transport engineering in South China University of Technology and have more than 3 years of experience in transport planning.

At each evaluation session, the system prototype was demonstrated and feedback was solicited through discussion and a formal questionnaire.

The results of the analysis of the questionnaire are shown in Table 3. Since the trip prediction model is designed for transport planner, the potential users were not required to answer question 8. The users were asked to use five-point scales (1=strongly disagree, 3=undecided, 5=strongly agree) to rate the effectiveness and usability of the prototype system. The potential users and transport planners rated the system with a mean score over 3.8 except the question 7. Why did the question 8 get a lower score? Most of the interviewees (transport planners) think the system may include more and more functions as DSS.

Table 3. Mean responses to the system evaluation by transport planner and potential user

Question	Transport Planner n=25		Potential user n=80	
	Mean rating[a]	S.D.	Mean rating[a]	S.D.
Effectiveness of the system				
1. Helps in selecting reasonable path	4.25	0.44	4.16	0.46
2. Provides clear information for selected path	4.37	0.42	4.27	0.41
Usability of the system				
3.System is easy to use	4.06	0.53	3.95	0.56
4.System is user friendly	4.00	0.61	3.97	0.62
5.System display is well designed	3.91	0.55	4.02	0.63
6.Response time in the system is acceptable	4.31	0.53	4.18	0.41
7.System contains functions which user requires	3.76	0.79	3.84	0.66
8.Accuracy of prediction is acceptable	4.04	0.57		

[a] Scale: 1=strongly disagree; 3=undecided; 5=strongly agree.

In the evaluation of trip planning system, further analysis is conducted to investigate whether there is a difference between transport planners and potential users except the last question. Can we conclude at the 5% significance level that a difference exists between them? The Wilcoxon signed rank sum test can be used to compare two populations when the data are quantitative (Keller [15]). We have two hypotheses:

H_0: The two population locations are the same;

H_1: The location of transport planner is different from the location of potential users.

It can be obtained from the analysis of the questionnaire: For the effectiveness, z_score = 1.34 and p_value=0.18. For usability, z_score = 0.539 and p_value=0.59. We cannot reject the null hypothesis. Therefore it can be concluded that there were no significant difference between transport planners and potential users at a 0.05 level of significance.

CONCLUSION

In this paper the research project on DSS for transit service planning is described. The DSS provides a flexible mechanism to deal with data. It may include various transit networks, such as networks in different cities, or networks at different time period in a city. Each transit network including data from different transit service provider in a specific time is stored in one unique database. User can select one database at present or in the future to analyze.

The prototype system was written using Visual Basic 6.0 and ran on a personal computer under a Microsoft Windows 98 environment. By developing and deploying the system, we have shown that it is a feasible procedure to use a recursive algorithm to find reasonable paths. The recursive algorithm for finding reasonable paths can also be useful in any other transit network. The database including information from different transit service providers may provide information as more as possible to travelers to plan their trips.

The prototype DSS works well in different scenarios. The responsive time of trip planning and trip prediction is acceptable in a Pentium III personal computer. The trip planning application will be completed immediately after the mouse is clicked. The training of FNN and trip prediction will be accomplished within one second.

A system evaluation was performed using expert validation, prototype testing and outcome evaluation to see whether the DSS achieved its designed purpose and functioned properly. The results of the questionnaires evaluation feedback strongly support the position that the DSS has achieved the primary objectives of being easy to use and able to enhance efficiency and effectiveness in supporting decision-making.

Based on the feedback from the potential users and experts, the following improvement to the prototype system are intended and will be added to a later version of the DSS.

An interactive trip planning application based on web-based will be developed, which must provide more and clear information than the trip planning program at present on the Web. A multi-language (Chinese, English) function will be incorporated in the user-interface. More models should be developed and added into it to enhance the ability to support transit service planning.

REFERENCES

[1] Arampatzis, G., Kiranoudis, C.T., Scaloubacas, P., et al. "A GIS-based decision support system for planning urban transportation policies," *European Journal of Operational Research*, Volume 152, Number 2, 2004,pp. 465–475.

[2] Benedetto, A. "A decision support system for the safety of airport runways_ the case of heavy rainstorms", *Transportation Research A*, Volume 36, Number 8, 2002, pp.665-682.

[3] Bielli, M. "A DSS Approach to Urban Traffic Management," *European Journal of Operational Research*, Volume 61, 1992, pp.106-113.

[4] Bielli, M., Ambrosino G. and Boero, M., *Artificial intelligence applications to traffic engineering*, VSP, Utrecht, The Netherlands, 1994.

[5] Chang, Y. H., Yeh, C. H. and Cheng, J. H. "Decision support for bus operations under uncertainty: A fuzzy expert system approach", *Omega International Management Science*, Volume 26, Number 3, 1998, pp. 367-380.

[6] Colorni, A., Laniado, E., and Muratori, S. "Decision support systems for environmental impact assessment of transport infrastructures", *Transportation Research D*, Volume 4, Number 1, 1999, pp.1-11.

[7] Desaulniers, G. and Villeaneuve, D. "The shortest path problem with time windows and linear waiting costs", *Transportation Science*, Volume 34, Number 3, 2000, pp.312-319.

[8] Ferrari, P. "A model of urban transport management", *Transportation Research B*, Volume 33, Number 1, 1999, pp. 43-61.

[9] Fierbinteanu, C. "A decision support systems generator for transportation demand forecasting implemented by constraint logic programming," *Decision Support Systems*, Volume 26, Number 1, 1999, pp.179–194.

[10] Gayialis, S.P. and Tatsiopoulos, I. P. "Design of an IT-driven decision support system for vehicle routing and scheduling", *European Journal of Operational Research*, Volume 152, Number 2, 2004, pp. 382–398

[11] Hall, R.W. "Route choice and advance traveler information systems on a capacitated and dynamic

network", *Transportation Research C*, Volume 4, Number 5, 1996, pp. 289-306.

[12] Hickman, M.D. and Wilson, N.H.M., "Passenger travel time and path choice implications of real-time transit information", *Transportation Research C*, Volume 3, Number 4, 1995, pp. 211-226.

[13] Ingmar Andreasson, "A method for the analysis of transit networks", In*: Advances in operations research*, Marc Rouberns ed., 1977.

[14] James, L. J., Database modes, language and design. Oxford university, New York,1997.

[15] Keller, G. and Warrack, B., *Statistics for management and economics*, Brooks Cole, Pacific Grove, 2000.

[16] Kengpol, A. "Design of a decision support system to evaluate the investment in a new distribution centre", *International Journal of Production Economics*, Volume 90, Number 1, 2004, 59-70.

[17] Kraus, M. and Yoshida, Y. "The Commuter's Time-of-Use Decision and Optimal Pricing and Service in Urban Mass Transit", *Journal of Urban Economics*, Volume 51, Number 1, 2002, pp. 170–195.

[18] Ledoux, C. "An urban traffic flow model integrating neural network", *Transportation Research C*, Volume 5, Number 5, 1997, pp.287-300.

[19] Moynihan, G. P., Pravin, S. R., Sterling, J. U., and Nichols, W.G., "Decision support system for strategic logistics planning", *Computers in industry*, Volume 26, Number 1, 1995, pp.75-84.

[20] Palacharla, P. and Nelson, P., "Application of fuzzy logic and neural networks for dynamic travel time estimation", *International Transactions in Operational Research*, Volume 6, Number 1, 1999, pp. 145-160.

[21] Peng, Z.R. and Huang, R.H. "Design and development of interactive planning for web-based transit information systems", *Transportation Research C*, Volume 8, Number 1-6, 2000, pp. 409-425.

[22] Shmueli, D. "Applications of neural networks in transportation planning", *Progress in Planning*, Volume 50, Number 3, 1998, pp.141-204.

[23] Tong, C. O. and Richardson, A. J. "A computer model for finding the time-dependent minimum path in a transit system with fixed schedules," *Journal of Advanced Transportation*, 1984, Volume 18, pp.145-161.

[24] Tong, C. O. and Wong, S. C. "A stochastic transit assignment model using a dynamic schedule-based network," *Transportation Research B*, Volume 33, Number 2, 1998, pp.107-121.

[25] Turban, E., *Decision support systems and intelligent systems*. Prentice-Hall, New Jersey, 1998.

[26] Yin, H.B., Wong, S.C., Xu, J.M. and Wong, C.K. "Urban traffic flow prediction using a fuzzy-neural approach", *Transportation Research C*, Volume 10, Number 2, 2002, pp. 85-98.

[27] Zhan, F.B. and Noon, C.E. "Shortest path algorithms: An evaluation using real road networks", *Transportation Science*, Volume 32, Number 1, 1998, pp.65-73.

[28] Zografos, K.G., Androutsopoulous, K.N., and Vasilakis, G.M. "A real-time decision support system for railway network incident response logistic", *Transportation Research C*, Volume 10, Number 1, 2002, pp.1-18.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous referees for their constructive and useful comments on the earlier draft of this paper. The research reported was supported by a research grant HKU 7018/99E from the Hong Kong Research Council and by the Natural Science Foundation of Guangdong Province, People Republic of China.

AUTHORS' BIOGRAPHIES

Man-chun Tan is at the College of Information Science and Technology, Jinan University, Guangzhou, 510632, People Republic of China.

Cong-on Tong is at the Department of Civil Engineering, The University of Hong Kong, Hong Kong, People Republic of China.

Jian-min Xu is at the College of traffic and communication, South China University of Technology, Guangzhou, People Republic of China.