XI Congreso de Ingeniería del Transporte (CIT 2014)

Shortest hyperpaths in a multimodal network for the public transportation system: Central Southern Mexico City

Minerva Soto Patiño*, Angélica Lozano

*Universidad Nacional Autónoma de México (UNAM), Institute of Engineering, Torre de Ingeniería Piso 2 ala Norte, Mexico City, 04510, Mexico.

Abstract

This work presents the implementation of an algorithm to obtain shortest hyperpaths in multimodal networks with a limited number of modal changes, for a real case in central-southern Mexico City. Five public transportations modes without fixed schedules are considered. The outcome for the user is a solution set (Pareto Optimal) with different travel times and number of modal changes, among which user can choose the most convenient for him/her. The implementation result is a set of files that can be used to provide information related to travel decision, and can be the base for creating an Advanced Traveler Information System.

Peer-review under responsibility of CIT 2014.

Keywords: Hypergraph, hyperpath, multimodal transport;

Nomenclature

- λ Path whose nodes represent the stops of a line
- A Set of lines of a public mode r
- λ i,j Line with stop at i

© 2014 The Authors. Published by Elsevier Ltd.
Selection and peer-review under responsibility of CIT 2014.

* Corresponding author. Tel.: +52-55-56233600, ext:8382
E-mail address: msotop@iingen.unam.mx
1. Introduction

In many organized cities, public transportation modes have schedules and the information to develop a shortest multimodal path model is available. In Mexico City, the public transportation system has modes without schedules and vehicles of some modes are not restricted to travel on a specific roads. Hence, it is difficult to know the exact places where lines pass and stop, the places where to do modal transfers, the waiting time at stops and then the time on a path.

This work introduces the implementation of an algorithm to obtain the shortest hyperpaths in multimodal networks, with a limited number of modal changes (Lozano and Storchi, 2002). This algorithm is applied to the real case of central-southern Mexico City, where several public transportations modes without schedules operate. Due to the lack of schedules, the time in between the user’s arrival at a stop and the vehicle’s arrival to the same stop is unknown. This time is estimated by means the hyperpaths algorithm. The considered modes are metro, BRT, light train, trolleybus and express lines of buses. The algorithm (Lozano and Storchi, 2002) includes a viability prove, which is not incorporated in this implementation due to the considered modes can be taken and left at any stop, without restriction on the sequence of modes on the path. The estimation of waiting times at stops as well as travel times are based on hypergraphs or hyper networks, which use line’s frequency of the considered modes. The outcome for the user is a Pareto Optimal set, composed of paths with different travel time and number of modal transfers, among which user can choose the most convenient one.

The implementation result is a set of files, containing shortest multimodal hyperpath information which can support user to take decisions on his/her trip, and can be the base for creating an ATIS (Advanced Traveler Information System).

The rest of the paper is as follows: first some basic concepts of hypergraphs and multimodal hypergraphs are presented, then the implementation of the multimodal hyperpaths algorithm is described, including the modified parts, then the case study is described, and finally the implementation is described, and conclusion is included.

2. Hyperpaths

Concepts on hypergraphs and multimodal hypergraphs are presented below.

2.1. Hypergraphs

Schettino & Pallottino (1999) and Lozano & Storchi (2002) present the theoretical base on hypergraphs and hyperpaths used in this work. Just some of these concepts as presented below.

A hypergraph or h-graph is a pair \( H = (N, E) \), where \( N \) is the set of nodes and \( E \) is the set of h-arcs. A h-arc \( e = (t(e), h(e)) \) is identified by its tail \( t(e) \in N \) and its head \( h(e) \subseteq N / t(e) \). If \( |h(e)| = 1 \), the h-arc is equivalent to an arc: \( e = (i, j) \).

In an h-graph, a path \( q_{od} \) connecting destination \( d \) and origin \( o \) is a sequence of nodes and h-arcs:

\[
q_{od} = (o = t(e_1), e_1, t(e_2), e_2, \ldots, e_m, d) \text{, where } t(e_i) \in h(e_i) \text{, to } i = 2, \ldots, m - 1, \text{ and } d \in h(e_m).
\]

A hyperpath \( p_{od} \) is the minimal acyclic set of paths \( q_{od} \), such that the destination \( d \), is connected to every node belonging to \( p_{od} \).

A stop-node is the focal point of public transport service, and the place where people board and leave vehicles.
To be able to understand all the aspects of how users employ this service, a detailed stop is shown in Fig. 1.

In view of the stop representation, an $h$-arc can be considered as a stop-node (tail) with the boarding lines set (head).

The attractive set is a priori chosen subset of lines to go from $i$ to $d$, such that, at node $i$, the user is willing to board the first vehicle of that subset which arrives.

A hyperpath from $o$ to $d$ ($p_{od}$) represents the set of possible paths for the user, where each boarding $h$-arc represents the attractive set for the user.

It is assumed that: passengers arrive randomly at each stop and always board the first vehicle of the attractive set which arrives; all lines are statistically independent, and each vehicle arrives to a stop with an exponential distribution.

A description of the expected waiting time at a stop, for the attractive set of lines, and the expected travel time on $h$-arcs and $h$-paths, is presented by Lozano & Storchi (2002).

In general, the arcs of a multimodal hyper-network have an associated travel time $C(e)=c(i,j)$ and a coefficient $\pi_j(e,j)=1$ since there are no other options to board at stop $i$. While a waiting time $C(e')=\omega(e')$ and as many coefficients as the number of nodes in $h(e')$, i.e. the coefficients $\pi_j(e',j) \forall j \in A'$, such that, $\sum_{j \in h(e')} \pi_j(e',j) = 1$ associated with each $h$-arc $e' = (i, h'(e'))$.

A value $V_p(i)$, which represents the expected travel time for going from $i$ to $d$, is associated with each hyperpath $p_{id}$, as shown in equations (1) and (2).

\[
V_p(i) = \begin{cases} 
  c(i,j) + V_p(j) & \text{if } e = (i,j) \\
  \omega(e') + \sum_{j \in h(e')} \pi_j(e',j)V_p(j) & \text{if } e' \text{ is a boarding } h \text{-arc} 
\end{cases}
\]

$V_p(o)$ is the expected travel time (of the whole trip) for a user who has the strategy for traveling represented by hyperpath $p$.

2.2. Multimodal hypergraph

The multimodal $h$-graph definition and the transport modes classification, given by Lozano & Storchi (2002), are below and in Table 1.

A multimodal $h$-graph is a triplet $H = (N, E, M)$ where $N$ is the set of nodes, $E$ is the set of $h$-arcs and $M$ is the set of modes associated with the $h$-arcs.
Table 1. Mode classification

<table>
<thead>
<tr>
<th>Subset</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>Public rail modes (Metro, light train)</td>
</tr>
<tr>
<td>M2</td>
<td>Public surface modes (BRT, RTP, trolleybus)</td>
</tr>
<tr>
<td>M3</td>
<td>Private modes with parking needs (cars and motorcycles)</td>
</tr>
<tr>
<td>M4</td>
<td>Private modes without parking needs (bicycle)</td>
</tr>
<tr>
<td>M5</td>
<td>Walking mode</td>
</tr>
<tr>
<td>M6</td>
<td>Modal transfer</td>
</tr>
</tbody>
</table>

A monomodal mode-$r$ $h$-graph, $r \in \mathcal{M}$ is represented by $H_r=(N_r,E_r)$ where $N_r$ and $E_r$ are the sets of nodes and $h$-arcs, respectively.

Only one mode-$r$ is associated with each $h$-arc of a multimodal $h$-graph. To obtain it, modal transfer arcs, i.e. arcs with the modal transfer "mode" associated, are added. Hence, an adjacent arc of a mode-$r$ arc is either a mode-$r$ arc or a modal transfer arc.

The considered modes in this implementation are M1, M2, M5 and M6.

3. Implementation of the Multimodal Hyperpath Algorithm

The Shortest Viable Hyper Path (SVHP) algorithm (Lozano and Storchi, 2002) works on multimodal networks, finding viable hyperpaths with the minimum expected time and no more than $k$ modal transfers. The solution is a Pareto-optimal set of hyperpaths for certain origin/destination pair, with minimum expected travel time and a number of modal transfers between 0 and $k$.

SVHP algorithm have a main procedure which calls arc-concatenation procedure and $h$-arc-concatenation procedure, to concatenate arcs and $h$-arcs, respectively. The algorithm also includes two procedures to verify viability of $h$-paths, called Determine-s and States procedures.

These last two procedures are not used in this implementation because the case study just includes modes which have not restrictions on their use, i.e., any modes sequence is possible. The considered M1, M2 and M5 modes can be taken and left at any stop.

3.1. Algorithm modification description

Main procedure of SVHP algorithm is shown in Fig. 2, including few changes presented in blue and red color. To avoid confusion some variables were renamed. The main change is the addition of an origin filter criteria, once a path to the origin $V^*(o)$ is found, all the paths that exceed in time $V^*(o)$ are omitted. Given that we consider a large real transport network, it is an important contribution because without this filter, the program continues making calculations, since the list $Q_{now}$ is not empty even if the solution is found, because the origin could be any node of the network.

Concatena-Arco Procedure (Fig. 3) was considerably reduced due to it is not necessary verify viability for this case study. The remainder procedure is like Dijkstra algorithm plus a modal transfer checking

Concatena-h-Arco Procedure (Fig. 4) has important modifications:
- Verifying viability is not necessary for this case study.
- A checking of $h$-arcs head sort is included, i.e. no $V^*(i)$ is calculated until all elements of $h(e)$ have a non-infinite $V^*$ assigned. This avoids random data input errors. For example, if by a reason one element of $h(e)$ that is not in the attractive set is calculated previously, and then introduced first in the procedure, it will throw a false attractive set, since the first element is never checked to be out of the set, and just added if it improves previous sets.
- A separated assignment of combined frequency $\Phi^*(e)$ and expected travel time $C^*(e)$ for one element and more than one is included. This has an important bearing on the Mexico City case since there are a lot of hyper arcs representing the approach to a line of rail transportation, such as metro, and it is important to
consider that for practical purposes these arcs are considered the same nature as the hyper arcs having an associated frequency, but an attractive set calculation is made to only consist of an element on the head node, i.e. \( |h(e)| = 1 \). In order to decrease time consuming of the program, \( h \)-arcs with \( |h(e)| = 1 \) are considered separately in the first section of the Concatena-\( h \)-Arco procedure.

### Procedure HRMC

1. begin
2.  foreach \( j \in N \) do
3.    \( V^*(j) = \infty \)
4.    lastlabel = \( \infty \)
5.    \( g(j) = 0 \)
6.  end
7.  foreach \( e \in E \) do
8.    \( C^*(e) = \infty; h^*(e) = \phi; \Phi^*(e) = 0; \)
9.  end
10. \( Q_{now} = \{d\}; V^*(d) = 0; \)
11. repeat
12.   while \( Q_{now} \neq \phi \) do
13.     Select&Remove(\( j, Q_{now} \))
14.     if \( (V^*(j) < \text{lastlabel}(j)) \) then
15.       if \( (V^*(j) < V^*(o)) \) then
16.         lastlabel(\( j \)) = \( V^*(j) \)
17.       foreach \( e \in B(j) \) do
18.         \( i = t(e) \)
19.         if \( (|h(e)| = 1) \) then
20.           call Procedure Concatena-Arco(\( e, V^*, c, g \))
21.         else
22.           call Procedure Concatena-\( h \)-Arco(\( e, V^*, C^*, g, h^*, \varphi \))
23.       end
24.     end
25.   end
26.   \( Q_{now} = Q_{next} \)
27.   \( Q_{next} = \phi \)
28.   \( z = z + 1 \)
29. until \( z > k \) or \( Q_{now} = \phi \);  
30. end

Fig. 2 Main procedure HRMC

### Procedure Concatena-Arco(\( e, V^*, c, g \))

1. \( wt = 0; \ con = 0; \)
2. if \( \text{modo}(e) = TF \) then
3.   if \( (g(j) < k) \) then \( wt = 1 \)
4. end
5. if \( (V^*(j) + c(e) < V^*(i)) \) then
6.   \( V^*(i) = V^*(j) + c(e) \)
7.  \( g(i) = g(j) + wt \)
8.  \( SA(i) = e \)
9. if \( (wt = 0 \ and \ \{i\} \notin Q_{now}) \ then \ Q_{now} = Q_{now} \cup \{i\} \)
10. if \( (wt = 1 \ and \ \{i\} \notin Q_{next}) \ then \ Q_{next} = Q_{next} \cup \{i\} \)
11.  \( \ con = 1 \)
12. end

Fig. 3 Procedure Concatena-Arco
3.2. Program

A Visual basic development is used to programming the algorithm, a string connection to a modified TransCad© dataview in .dbf extension is used to read the data. Any Geographical Information System (GIS) can be used instead of TransCad©. A set of data records is created, for each run of the program. The program consists of a main procedure and 12 auxiliary functions, one of them is a hyper path recovery process.

3.3. Outcome

The outcome of the program is a set of .csv files, a file for each member of the Pareto-optimal set. Each file has five columns:
- Memory Id from nodes list
- Node Id from TransCad© or another GIS.
- Travel time from that node to destination.
- Mode
- Arc Id from TransCad© or another GIS (node and subsecuent node).

The resulting file can be opened in a GIS, in order to visualize the found shortest hyper-path (using the Arc Id field in “select by value” selection).
4. Case Study

The case study of a part of Mexico City is described below.

4.1. Public transport in Mexico City

In Mexico City the public transport can be offered by the government or by private companies with governmental authorization. Just the first group is considered in this work, and it is composed of metro (ME), BRT (MB), light train (TL), trolleybus (TLB), and RTP bus service; here only express RTP service is taken into account.

Since the lack of information on the second group, these modes are not included. However, one of the modes offered by the government (RTP) has similar characteristics than the modes of the second group; hence they could be included later.

4.2. Study area

The study area was chosen so that most modes of public transportation in the city were covered. The study area contains the lines of public transportation modes, which allows to show the functionality of the algorithm and to provide an idea about the extension and complexity of public transportation in Mexico City.

The algorithm implementation for this study area can be the base of an Advanced Traveler Information System (ATIS), which can cover the whole Mexico City and includes all the modes.

The area covers over 171 km², which represents 21.7% of the urbanized area of the Mexico City consisting of approximately 788 km² (INEGI 2010).

The study area is shown in Fig. 5 on Google™ Earth, and its location within Mexico City is shown in green color in Fig. 6.

4.3. Modes modelling

Both modes M1 and M2 are modelled as hypergraphs, given they do not have schedules. In this case the metro mode can be used more than once and continue been viable, hence there is no need to prove viability as considered by SVHP algorithm. The other modes can be boarded and left at any stop, hence to prove viability is not necessary.
5. Implementation

Once the study area was defined and also were determined the modes to be considered as well as the algorithm to be based on, the next step was creating the corresponding hyper network. This gave a database which can be consulted by the algorithm, and has a geographical representation of the study area. The tools used for the algorithm implementation are the following: Google Earth for digitalizing data, ArcMap™ and Quantum GIS for converting digital data to shape files, and a GIS (TransCad©) to edit and complete the hyper network topology. Also Visual Basic development environment is used to implement the algorithm.

5.1. Data digitalizing

Google™ Earth was used for digitalizing the networks. A .kml file was created for each mode, and this file included all the stops and lines of the mode. Position mark and route tools were used for this labor.

The following step was to create a shape file. It can be done by means Xtools Pro using ArcMap™ or by using Quantum GIS; for the second one, two .kml files per mode are required, one containing stops and another containing routes. When all the information captured in Google™ Earth have shape format, it can be opened in a GIS to edit and build the needed topology for the hyper network.

5.2. Monomodal hyper networks edition

Mode by mode, each shape file was opened and exported as “Standar Geographic File” (.dbd). The .dbd file was opened and modified in order to add the required fields. Table 2 shows the structure used. To recreate the $h$-arc representation for each stop (see Fig. 1), the information related to arcs (travel time) and $h$-arcs (frequency) was added as shown in Fig. 7.
5.3. Networks joining

Once all the monomodal hyper networks are properly edited, it was necessary to join all of them in just one layer and to create the modal transfer arcs which connect them. The main characteristics of the final multimodal hyper-network, for this case study, is shown in Table 3 and the final multimodal hyper network is presented in Fig. 8.

Table 2 Attribute table fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Length</th>
<th>Decimals</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINEA</td>
<td>Transportation line of the corresponding mode</td>
<td>Character</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>MODO</td>
<td>Transportation mode</td>
<td>Character</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>RUTA</td>
<td>Route</td>
<td>Character</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>FREC_H_PIC</td>
<td>Rush hour frequency</td>
<td>Real (8 bytes)</td>
<td>19</td>
<td>11</td>
</tr>
<tr>
<td>FREC_H_VAL</td>
<td>Off-peak hour frequency</td>
<td>Real (8 bytes)</td>
<td>19</td>
<td>11</td>
</tr>
<tr>
<td>T_D_REC</td>
<td>Travel time</td>
<td>Real (8 bytes)</td>
<td>19</td>
<td>11</td>
</tr>
<tr>
<td>TIPO</td>
<td>Arc type</td>
<td>Character</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3. Characteristics of the final hyper network

<table>
<thead>
<tr>
<th>Arc type</th>
<th>Quantity</th>
<th>Length (km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A: arc</td>
<td>1183</td>
<td>414.15</td>
</tr>
<tr>
<td>B: transboard</td>
<td>11</td>
<td>1.72</td>
</tr>
<tr>
<td>H: h-arc</td>
<td>602</td>
<td>1.34</td>
</tr>
<tr>
<td>P (TF): pedestrian arc (modal transfer)</td>
<td>312</td>
<td>61.04</td>
</tr>
<tr>
<td>Total</td>
<td>2108</td>
<td>478.25</td>
</tr>
</tbody>
</table>

Fig. 7 Network edition
6. Results

The result generated by the program provides minimal hyper paths, sorted by the number of modal transfers, so that they can be from 0 to $k$. These hyper paths form a Pareto Optimal set. The solutions provided by this algorithm, unlike existing programs, considers the waiting time before boarding a public transport vehicle, and it is not offers just a path. Instead, the user can obtain a set of hyperpaths, from where he/she can choose the best according the preferred travel time and number of transfers.

The algorithm takes about two minutes per query (on a Windows 2007 64-bit OS, Core i7 3.4 GHz processor with 8GB of RAM), an acceptable time considering the size of the network and that it is a pseudo polynomial problem. This time can be reduced by improving the functions, working on a more powerful computer programming on a lower level language.

The presented implementation is an attempt to provide a real application which can be the base for an ATIS (Advanced traveler Information Systems). The implementation includes a database and a hyper-network; they jointly with the .csv file, could be used to provide information to users.

This would be an important advance in our country since, given that only a theoretical attempt made by the Mexican Institute of Transport by (Acha Daza & Espinosa Rescala 2004) to define a national architecture for ITS (Intelligent Transportation Systems), and just one real hyper-paths application, for the UNAM Campus, was done by López et al. (2013).

References