Computer and Information Science
www.ccsenet.org/journal.html
Vol. 1, No. 1
February 2008

Speedy Algorithm of Public Traffic Route Selection Based on Adaptive Backbone Network

Xiaowen Xu
Management School, Xi'an Jiaotong University, Xi'an 710049, China
E-mail: xiaowen@mail.xjtu.edu.cn
Jiayin Wang
School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China
E-mail: wangjiayin@mail.xjtu.edu.cn

The research is supported by National Natural Science Foundation of China (No. 70702030) and National Undergraduate Innovation Experimental Project of China (No. 610762).

Abstract

The inquiry service of public traffic routes is the important part of urban information service, which core is the public traffic route selection algorithm. However the traditional public traffic route selection algorithms have high time complexities and cannot support the inquiry of multiple changes. In this article, we put forward the speedy algorithm of public traffic route selection based on adaptive backbone network. In this algorithm, if the public traffic routes which pass certain public traffic station exceed or equal a certain value, so the station is defined as the backbone station, and the backbone stations and the public routes which pass them compose backbones network. If changes are limited at the backbone stations, so we can realize multiple changes inquiry and reduce computation in 10% of traditional algorithms through changing the certain value to adjust the backbone network.

Keywords: Adaptive backbone network, Backbone stations, Tabu Search, Public traffic route selection, Speedy algorithm

1. Introduction

The inquiry service of public traffic routes is the important measure to convenient for people's daily life and enhance the operation efficiency of public traffic route network, and it is also the important composing of urban information service, which core is the public traffic route selection algorithm. At present, traditional public traffic route selection algorithms include Dijkstra algorithm, Floyd algorithm or their ameliorations (Lu, 2001, p.68-70, Wang, 2007, p.63-67, Yue, 1999, p.209-212, Zhu, 2007, p.121-122 & Xu, 2005, p.16-17), which time complexities respectively are $O(|V|^2)$ and $O(|V|^3)$ (Thomas, 2007, p.324-330), but when a large of public traffic stations exist, higher time complexity will induce that inquiry time far exceeds what inquirer hopes. At the same time, these two algorithms cannot support route inquiry of multiple changes and cannot realize the first aim of least change times (Yang, 2000, p.87-91 & Ma, 2004, p.38-44). Aiming at problems existing in traditional algorithms, in this article, we put forward the speedy algorithm of public traffic route selection based on adaptive backbone network. This algorithm induces computation quantity and expedites computation speed through the backbone network technique, and can support the route inquiry of multiple changes through the adaptability of backbone network.

2. Adaptive backbone network

Suppose that the public traffic stations are the crunodes in the public traffic network map and all public traffic routes between any two public traffic stations are the directional routes, so the public traffic network map can be noted as $G = (V, E, L)$, where, V is the set of all public traffic stations, E is the set of all directional public traffic routes between any two public traffic stations, and L is the set of all public traffic routes.

Taking the public traffic route layout of the main city zone in Beijing, the main city zone in Beijing has 3957 public traffic stations (No.0001-No.3957) and 520 public routes (No.001-No.520). Suppose that v_i represents the public traffic station which number is i, l_j represents the public traffic route which number is j, so $V = \{v_1, v_2, \cdots, v_{3957}\}$ and $L = \{l_1, l_2, \cdots, l_{520}\}$. And if station v_i and station v_j ($i, j \in [1,3957]$) are located on the uplink (or downlink) at the public route l_k ($k \in [1,520]$) and the next station of v_i is v_j, so the directional route (i, j) is one side in map G.

Every public traffic route must pass some pubic traffic stations, and every public traffic station must at one public traffic route at least, and suppose that the set of all public traffic routes which pass station v_i is $L(v_i)$, so the set of all stations which pass station v_i on all the public traffic routes is $V(L(v_i))$.

Suppose that the station v_i is the backbone station with n adaptive degree which fulfills $L(v_i) \geq n$ ($n \in N$) and is noted as v_i^n, $V^n = \{v_i^n \mid i \in [1,3957] \wedge L(v_i) \geq n\}$ is the set of all backbone stations with n adaptive degree, E^n is the set of all directional public traffic routes between any two backbone stations with n adaptive degree, L^n is the set of public routes which pass the backbone station with n adaptive degree, and the sub-map of the public traffic network map which only includes all backbone stations with n adaptive degree is the backbone network with n adaptive degree, which is noted as $G^n = (V^n, E^n)$.

To establish the backbone network with n adaptive degree should possess two conditions, one is which can cover most public traffic routes and the other is that the purpose station which is reachable on the public traffic route also can be reachable on the backbone network with huge probability. Suppose that the reachable probability η between any two stations on the backbone network is the square of the ratio between the sum of station which L^n covers and the gross of public traffic stations, so Table 1 can be obtained through computation.

From Table 1, when $n = 20$, $|V^n| = 335$ and $L^n = 505$, i.e. the backbone stations which only occupy 8.47% of the gross of public traffic station cover 97.12% of public traffic routes, and the reachable probability between any two stations on the backbone network achieves 96.25%. Therefore, the reachability among most stations can be actualized through the backbone network and changes among backbone stations, and suppose that all changes can be actualized only at backbone stations in this article, thus it only needs to compute backbone stations not all public traffic stations, and when the gross of public traffic stations is large, this method can effectively reduce computation quantity and get approximate optimal solution. Suppose that the limitation of inquiry may induce the route obtained is not the optimal route, i.e. the approximate optimal solution, and considering the actual situations that the backbone stations are the zones that ten or tens of public traffic routes gather and the zones with dense human streams, so the route strange inquirer gets is convenient for memory and identification. If the station inquired cannot be reached through the backbone network, the backbone network needs to be extended. The so-called adaptive backbone network is to dynamically control the scale of backbone network according to actual needs of inquiry, which can not only ensure the solvability, but also achieve the speediness of solution.

The basic idea to establish adaptive backbone network algorithm is that the inquirer gives that start station v_{start} and the purpose station v_{end}, and if $n = \min(L(v_{start}), L(v_{end}))$ and go through all public traffic stations, the station v_i which fulfills $L(v_i) \geq n$ is the backbone station with n adaptive degree, so the backbone network $L(v_i) \geq n$ is confirmed.

3. Algorithm description

The basic idea of route selection algorithm based on the backbone network is that inquirer gives any start station v_{start} and purpose station v_{end}, firstly compute $L(v_{start})$ and $L(v_{end})$, judge whether $L(v_{start}) \cap L(v_{end})$ is ϕ, and if it is not ϕ, so v_{start} and v_{end} can be reached directly, so record the reachable public traffic route, or else v_{start} and v_{end} can not be reached directly, and the change is needed.

If v_{start} and v_{end} can not be reached directly, we should consider one time change, compute the set $V_B(L(v_{start}))$ of all backbone stations on those public traffic routes which pass the station v_{start} and the set $V_B(L(v_{end}))$ of all backbone stations on those public traffic routes which pass the station v_{end}, judge whether $V_B(L(v_{start})) \cap V_B(L(v_{end}))$ is ϕ, and if it is not ϕ, so v_{start} and v_{end} can be reached through one time change at some backbone station, so record the reachable public traffic route with one time change, or else v_{start} and v_{end} can not be reached through one time change, and multiple changes are needed.

If v_{start} and v_{end} can not be reached through one time change, we should consider two times changes, firstly compute the set $V_{start} = V_B(L(v_{start}))$ of backbone stations that the start station can reach directly, then compute the set $V_{end} = V_B(L(v_{end}))$ of backbone stations which can reach the purpose station directly, and let $v_i \in V_{start}$ and $v_j \in V_{end}$ at random, judge whether v_i and v_j can be reached directly, i.e. look for l_k and make $v_i \in V(l_k) \cap v_j \in V(l_k)$, and if one or more l_k can be found, it indicates v_{start} and v_{end} can be reached through two times changes, so record the reachable public traffic route with two times change, or else v_{start} and v_{end} can not be reached through two times change.

If v_{start} and v_{end} can not be reached through two times change, i.e. any station in v_{start} and any station in v_{end} can not be reached directly, we can consider three times changes, and for this situation the recursion is used. Let $v_i \in V_{start}$ and $v_j \in V_{end}$ at random, judge whether v_i and v_j can be reached directly, and the algorithm that computes whether two stations can be reached through one time change has been given. If v_i and v_j can be reached through one time change, it indicates that v_{start} and v_{end} can be reached through three times changes, and if any station in v_i and any station in v_j can not be reached through one time change, so it indicates that v_{start} and v_{end} can not be reached through three times changes. Through the recursion in turn, any two stations can be reached through some times changes.

The steps of the algorithm are as follows:

The inputs of the algorithm include the start station v_{start} and the end station v_{end}. The outputs of the algorithm include the sets of reachable routes.

Step 1: Suppose that cyclic variable $i=1$, variable $L(v_{start}) = L(v_{end}) = \phi$, take l_i in L, and if $v_{start} \in V(l_i)$, so $L(v_{start}) = L(v_{start}) \cup \{l_i\}$, and if $v_{end} \in V(l_i)$, so $L(v_{end}) = L(v_{end}) \cup \{l_i\}$, and if $i \neq 520$, so $i = i+1$ and go to Step 1.

Step 2: If $L(v_{start}) \cap L(v_{end}) \neq \phi$, so record these reachable routes and exit.

Step 3: Suppose that cyclic variable $i=1$, variable $V_B(L(v_{start})) = \phi$, take l_i in $L(v_{start})$, and let $V_B(L(v_{start})) = V_B(L(v_{start})) \cup \{v_j \mid v_j \in V(l_i) \cap v_j \in V^n\}$ (i.e. merge all backbone stations on route l_i into $V_B(L(v_{start}))$, where V^n is confirmed by the algorithm established by the backbone network.), if $i < |L(v_{start})|$ and $i = i+1$ then go to Step 3.

Step 4: Suppose that cyclic variable $i=1$, variable $V_B(L(v_{end})) = \phi$, take l_i in $L(v_{end})$, and let $V_B(L(v_{end})) = V_B(L(v_{end})) \cup \{v_j \mid v_j \in V(l_i) \cap v_j \in V^n\}$ (i.e. merge all backbone stations on route l_i into $V_B(L(v_{end}))$, where V^n is confirmed by the algorithm established by the backbone network.), if $i < |L(v_{end})|$ and $i = i+1$ then go to Step 4.

Step 5: If $V_B(L(v_{start})) \cap V_B(L(v_{end})) \neq \phi$, so record these reachable combined routes through one time change and exit.

Step 6: Take v_i in $V_B(L(v_{start}))$ at random, and take v_j in $V_B(L(v_{end}))$ at random, let $v_{start} = v_i$ and $v_{end} = v_j$, i.e. take v_i and v_j as the new start station and the new purpose station, recursively transfer the algorithms in Step 1 and Step 2, and if v_i and v_j can be reached directly, so v_{start} and v_{end} can be reached through two times changes, export route selection information and exit.

Step 7: Take v_i in $V_B(L(v_{start}))$ at random, and take v_j in $V_B(L(v_{end}))$ at random, let $v_{start} = v_i$ and $v_{end} = v_j$, i.e. take v_i and v_j as the new start station and the new purpose station, recursively transfer the algorithms in Step 3, Step 4, and Step 5, and if v_i and v_j can be reached directly, so v_{start} and v_{end} can be reached through three times changes, export route selection information and exit, recursively transfer until the feasible solution with least change times can be obtained.

The algorithm will get some feasible route combinations with "least change times", so we can evaluate these routes according to inquirer's actual needs and select one group or several groups of optimal route combinations which can fulfill inquirer's requirement.

4. Conclusion

In this article, we put forward the speedy algorithm of public traffic route selection based on adaptive backbone network. In this algorithm, if the public traffic routes which pass certain public traffic station exceed or equal a certain value, so the station is defined as the backbone station, and the backbone stations and the public routes which pass them compose the sub-map of public traffic network map, i.e. the backbones network. The algorithm supposes that change can only be actualized at the backbone station, and because backbone stations almost cover all pubic traffic routes, so most public traffic stations can be reached under that hypothesis. Because the quantity of backbone stations generally don't exceed 10% of the public traffic station gross, so the algorithm only needs to compute backbone stations not all public traffic stations when computing changes, and the computation quantity can be reduced fully comparing with traditional algorithms, and the speedy inquiry can be actualized. At the same time, the algorithm can actualize the inquiry with multiple changes through recursive computation, which can better fulfill inquirer's demand. Considering the solvability of the route inquiry among any public traffic stations, this article puts forward the method to dynamically control the scale of backbone network, i.e. adaptive backbone network, accordingly gives attention to the solvability and the speediness of the solution.

References

Lu, Zhenbo & Huangwei. (2001). City Public Transport Guiding System: Applications of GIS in Public Transport Management. *Journal of Highway and Transportation Research and Development.* Vol.18. No.5. p.68-70.

Ma, Lianghe, Liu, Xinbin & Liao, Daqing. (2004). The Shortest Path of Public Transport Net and Related Optimal Taking Bus Problem. *Mathematics in Practice and Theory.* Vol.34. No.6. p.38-44.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest & Clifford Stein. (2007). *Introduction of Algorithms (2nd Edition).* Beijing: Machine Industry Press. p.324-330.

Wang, Shixiang & Rao, Weiya. (2007). The Data Structure of Inquiry for Public Transport Route and the Realization of Its Algorithm in Major City. *Computer Systems Applications.* No.9. p.63-67.

Xubing & Xie, Shiyi. (2005). The Implementation of Bus Change Algorithm Based on Station Priority. *Fujian Computer.* No.7. p.16-17.

Yang, Xinmiao, Wanghui & Ma, Wenteng. (2000). GIS-Based Public Transit Passenger Route Choice Model. *Journal of Southeast University (Natural Science Edition).* Vol.30. No.6. p.87-91.

Yueyang & Gong, Jianya. (1999). An Efficient Implementation of Shortest Path Algorithm Based on Dijkstra Algorithm. *Journal of Wuhan Technical University of Surveying and Mapping (WTUSM).* Vol.24. No.9. p.209-212.

Zhu, Jiangyun & Wang, Yukun. (2007). The Optimal Route Algorithm Based on Least Change Times. *Fujian Computer.* No.3. p.121-122.

Symbol table.

V	The set of all public traffic stations
v_i	The public traffic station which number is i
L	The set of all public traffic routes
l_j	The public traffic route which number is j
$\|A\|$	The degree of set A, A is the set of public traffic stations or routes
$V(A)$	The set of all public traffic stations at A, A is the public traffic station or the set of public traffic station
$V_B(A)$	The set of all backbone stations at A, A is the public traffic station or the set of public traffic station
$L(v_i)$	The set of public routes which pass station Vi
v_i^n	The backbone station with n adaptive degree which number is i
V^n	The set of all backbone stations with n adaptive degree
L^n	The set of public routes which pass the backbone station with n adaptive degree
E^n	The set of all directional public traffic routes between any two backbone stations with n adaptive degree

v_{start}	The start station which inquirer gives
v_{end}	The purpose station which inquirer gives

Table 1. The reachable probability of any two stations in the backbone network

n	$\|V^n\|$	L^n	η^n
10	846	516	99.14%
11	807	516	99.14%
12	704	515	99.14%
13	657	515	99.14%
14	559	514	99.04%
15	528	513	98.64%
16	466	510	97.24%
17	442	510	97.24%
18	398	508	96.74%
19	373	505	96.25%
20	335	505	96.25%