

距离加权公交换乘复杂网络最短路算法研究

Shortest Path Algorithm Based on the Weighted Distance in Complex Bus Transfer Network

郑健琛¹, 陈建宇¹, 龙燕君²

(1.西南交通大学交通运输与物流学院,四川 成都 610031;2.西南交通大学艺术与传播学院,四川 成都 610031)

Zheng Jianchen¹, Chen Jianyu¹, Long Yanjun²

(1. School of Transportation and Logistics, Southwest Jiaotong University, Chengdu Sichuan 610031, China; 2. School of Art and Communication, Southwest Jiaotong University, Chengdu Sichuan 610031, China)

摘要: 为研究乘客使用公共交通的实际出行距离,基于公交复杂网络中的换乘网络Space P拓扑结构,结合公交车站的经纬度坐标,建立以距离为边权的加权公交换乘网络。基于该加权网络,设计了综合考虑换乘次数和路径长度的最短路算法,该算法可保证在站间换乘次数最少的基础上通过的路径也相对最短。利用成都市公交网络进行实例分析,并与Floyd算法进行对比,结果显示,由该算法得到的平均最短路径长度增加3.7 km,但平均换乘次数下降0.64次,更符合乘客的出行习惯;随机选择一些车站进行最优换乘路径选取试验,结果表明,由该算法得到的方案在保证换乘次数最少基础上,得到的路径也基本最短,证明了算法的有效性。

Abstract: In order to study the actual travel distance of passengers using public transportation, this paper develops a distance-weighted complex bus transfer network with the geodetic coordinates of bus stops based on the Space P topological mapping method in the complex network theory. On the basis of the weighed network, the paper further develops a shortest path calculation algorithm, which guarantees that the path is the shortest based on the premise of having the minimum transfer times. Then, data from Chengdu public transportation network is used to validate the proposed algorithm. Comparison with the Floyd algorithm indicates that the proposed algorithm has slightly increased the average shortest distance by 3.7 km but decreases the average transfer by 0.64 times, which is more in agreement with travel habits of passengers. Finally, the effectiveness of the algorithm is verified by experiments with randomly selected bus stops. The results show that a shortest path with minimum transfer times can be obtained with the proposed algorithm.

关键词: 复杂网络; 公交换乘; 最短路算法; 距离加权

Keywords: complex network; bus transfer; shortest path algorithm; weighted distance

中图分类号: U491

文献标识码: A

收稿日期: 2012-05-17

作者简介: 郑健琛(1987—),男,四川成都人,硕士研究生,主要研究方向:城市交通复杂网络特性。E-mail:zjc0506@me.com

0 引言

城市公交网络由大量的车站、线路交织而成,是中国大部分城市最主要的公众出行载体。研究表明,公交网络是一个具有典型小世界特性的复杂网络^[1-3]。根据对公交乘客进行的心理调查统计结果,换乘次数最少是其出行时考虑的首要因素。目前,一些利用复杂网络求解公交车站间最短路径的方法,在进行路径选择时仅考虑了优先选择换乘次数少的路径方案,并不能对比换乘次数相同时各方案的优劣^[4]。

本文利用车站经纬度坐标得到站间距离,构建以距离为网络边权的加权公交复杂网络。基于该加权网络,设计了综合考虑换乘次数和路径长度的最短路算法,从而保证在站间换乘次数最少的基础上通过的路径也最短。

1 两种公交网络拓扑映射方法

构造城市公交系统的复杂网

络需要考虑网络拓扑的连通关系。文献[5]在对波兰 22 个公交系统进行复杂网络特性分析统计过程中, 采用两种方法建立公交网络拓扑, 即根据地理空间关系的车站网络 Space L 和根据换乘空间关系的换乘网络 Space P。

1.1 车站网络 Space L

车站网络 Space L 是将公交车站视为节点, 若两个车站是一条或多条线路上的相邻车站, 则这两个车站相连。这种网络拓扑的构造方式与实际公交系统中车站的地理空间关系相似, 见图 1a, 公交车站网络由 7 个车站(S1, S2, S3, S4, S5, S6, S7)组成, 包含 4 条线路, 分别为线路 1(S1—S2—S3—S4)、线路 2(S2—S6)、线路 3(S3—S5)和线路 4(S5—S6—S7)。

1.2 换乘网络 Space P

换乘网络 Space P 的构建也是以车站为节点, 若两个车站间有一条或多条公交线路通过, 无论两车站在线路中的地理位置是否相邻, 两车站间都存在连边。将图 1a 中的网络表示成 Space P 的形式就如图 1b 所示, 连边表示从其中一个车站能够

不换乘而直达另一车站, 任意两站间的最短路径表示换乘次数最少的乘车方案, 例如从 S1 到 S6 的最短路径为 S1—S2—S6, 中间在 S2 处换乘 1 次。由于构建的 Space P 网络通常都是无权重的, 在最优路径选择中很可能存在多条最短路径。由 Space P 网络可以得出从 S1 到 S7 共存在两条最短路径, 方案 1(S1—S2—S6—S7)和方案 2(S1—S3—S5—S7), 这两个方案均换乘 2 次, 但根据方案各自经过的车站数量, 可以看出方案 1(3 站)要优于方案 2(5 站)。鉴于 Space P 网络的这种缺陷, 考虑在网络连边中引入距离权重, 构建加权的 Space P 网络, 确保在换乘次数最少的情况下, 路径长度也最优。

2 加权换乘网络的建立

在 Space P 网络中, 连边的权值等于从一个车站经一条线路行驶至另一车站的总距离。公交车辆的行驶路线一般为曲线, 其距离往往难以得到, 但线路中相邻两车站间的行驶路径可以近似认为是直线, 其距离可以利用车站的经纬度坐标用式(1)求得^[6]。

$$a_{12} = \sqrt{\{(Long_1 - Long_2)\pi R \cos[(Lat_1 + Lat_2)\pi R / 360^\circ] / 180^\circ\}^2 + [(Lat_1 - Lat_2)\pi R / 180^\circ]^2}, \quad (1)$$

式中: a_{12} 为车站网络中两相邻车站间的近似直线距离; $Long_1$, $Long_2$, Lat_1 , Lat_2 分别为两车站的经度和纬度; R 为地球半径, 取 6 370 km。

得到线路中相邻车站间的距离后, 可将换乘网络 Space P 中任意有连边的车站 i, j 间连边的权重 w_{ij} 描述为: 经过 i, j 车站的所有线路中, 从 i 站到 j 站行驶路径最短的线路距离。

$$w_{ij} = \min(p_{ij}^1, p_{ij}^2, p_{ij}^3, \dots, p_{ij}^n), \quad (2)$$

$$p_{ij}^l = \sum_{s=i}^{j-1} a_{s,s+1}^l, \quad (3)$$

式中: p_{ij}^l 为经过 i, j 车站的第 l 条线路从 i 到 j 的距离; $a_{s,s+1}^l$ 为第 l 条线路从 i 车站起依次经过的车站间距, 即第 l 条线路上车站 s 与其下一车站 $s+1$ 的距离。若 i, j 两站间无连边存在, 说明两站

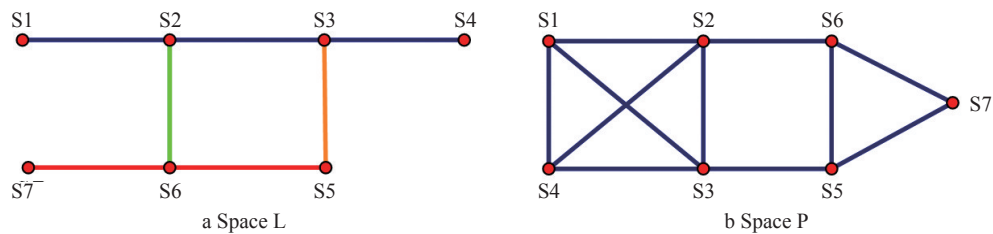


图 1 Space L 网络与 Space P 网络示意图

Fig.1 Illustration of Space L and Space P networks

间需要通过换乘才能到达, 此时 $w_{ij} = \infty$ 。

如图2所示, i 站和 j 站间有 p_1 和 p_2 两条线路直达, 假设各站间距 a_i 相同且均为1, 则 $w_{ij} = \min(p_1, p_2) = \min(a_1 + a_2 + a_3, a_4 + a_5) = 2$ 。显然, 当车站 i, j 为一条线路的两相邻车站时, $w_{ij} = a_{ij}$ 。

3 加权网络最短路径求解

根据第2章建立的加权网络, 得到任意有直达公交线路经过的车站间最短路径, 即 Space P 网络连边的权值。若公交网络中没有孤立的车站和线路, 利用传统的最短路求解算法 Dijkstra 和 Floyd 也能得到该网络的最短路径, 但是由此获取的最短路径仅考虑了车辆行驶路径最短, 乘客在出行过程中往往需要多次换乘, 并不具备实用性。本文将提出一种算法, 在任意车站间换乘次数最小的基础上, 给出车辆行驶路径也相对最短的乘车方案。

算法思路和假设: 1) 假定乘客利用公交出行时首先考虑乘车次数最少, 其次考虑乘车路径最短; 2) 将 r 次换乘才能到达的车站对之间的换乘过程拆分为: 先经过 $r-1$ 次换乘, 然后再经过1次换乘完成。然后, 可按如下思路计算换乘次数最小时最短路径的大小。

1) 根据 Space P 网络的拓扑结构建立对应的可达矩阵 R , 若 i, j 两站间有边相连, 则 $r(i, j) = 1$, 否则 $r(i, j) = \infty$ 。当 $i = j$ 时, $r(i, j) = 1$ 。

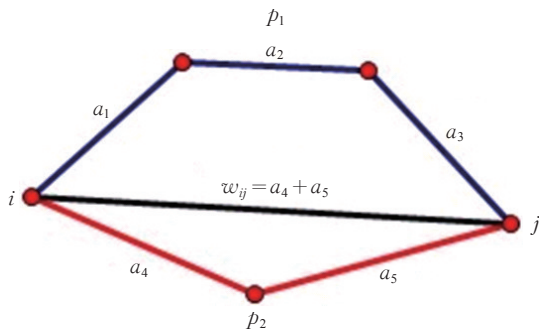


图2 距离权重确定示意图

Fig.2 Illustration of the determination of distance-weight

2) 在可达矩阵 R 的基础上, 利用 Dijkstra 算法或 Floyd 算法得到 R 的最小乘车次数矩阵 D 。根据 Space P 网络的定义, 矩阵 D 中的元素 d_{ij} 代表从 i 站到 j 站的最少乘车次数^[7]。

3) 根据第2章所得结果, 构建 Space P 网络的加权距离矩阵 W_0 , W_0 为该网络中不需换乘(乘车次数 $d_{ij} = 1$)便能直达的车站对间的最短路径长度。

4) 由 Dijkstra 最短路算法原理可知, 需换乘 $r (r \geq 1)$ 次的站间最短路径长度, 可拆分为前 $r-1$ 次换乘的最短路径长度加上第 r 次换乘到达终点的最短路径长度。因此, 最小乘车次数为 $d_{ij} = r + 1 (d_{ij} \geq 2)$ 的车站对 i, j 之间的最短路径长度可以表示为经过 $r-1$ 次换乘的最短距离 W_{r-1} , 再加上第 r 次换乘距离的总和。查矩阵 D , 对于 D 中不同换乘次数的车站, 分别建立对应次数的最小换乘距离矩阵 $W_1, W_2, W_3, \dots, W_r$, 建立规则为

$$\begin{cases} w_{ij}^r = \min(w_{ik}^{r-1} + w_{kj}^0), & d_{ij} = r + 1, \\ w_{ij}^r = \infty, & d_{ij} \neq r + 1, \end{cases} \quad (4)$$

式中: k 为第 r 次换乘的车站, 同时也是第 $r-1$ 次换乘的终点站。

5) 将算出的所有换乘距离矩阵以式(5)所示的规则整合为一个矩阵 W 。 W 为该公交网络任意两站间保证换乘次数最少情况下的最短路径矩阵, 算法结束。

$$w_{ij} = \min(w_{ij}^0, w_{ij}^1, \dots, w_{ij}^r). \quad (5)$$

以图1为例, 首先建立 Space P 网络的可达矩阵 R 和 R 的最小乘车次数矩阵 D , D 中最大的元素为3, 说明该网络任意两站间最大换乘次数为2。然后, 根据第2章所得的结果建立 Space P 网络的加权距离矩阵 W_0 。查矩阵 D , 对于 $d_{ij} = 2$ 的车站对, 根据式(4)建立一次换乘距离矩阵 W_1 , W_1 为网络中需要换乘1次的两站间的最短路径; 对于 $d_{ij} = 3$ 的车站对, 继续建立二次换乘距离矩阵 W_2 , 此时已建立了该网络在最大换乘次数下的距离网络。利用式(5)将 W_0, W_1, W_2 整合, 得到该公交网络任意两站在换乘次数最少情况下的最短路径矩阵 W 。

$$R=\begin{bmatrix}1&1&1&1&\infty&\infty&\infty\\1&1&1&1&\infty&1&\infty\\1&1&1&1&1&\infty&\infty\\1&1&1&1&\infty&\infty&\infty\\\infty&\infty&1&\infty&1&1&1\\\infty&1&\infty&\infty&1&1&1\\\infty&\infty&\infty&\infty&1&1&1\end{bmatrix},$$

$$D=\begin{bmatrix}1&1&1&1&2&2&3\\1&1&1&1&2&1&2\\1&1&1&1&1&2&2\\1&1&1&1&2&2&3\\2&2&1&2&1&1&1\\2&1&2&2&1&1&1\\3&2&2&3&1&1&1\end{bmatrix},$$

$$W_0=\begin{bmatrix}0&1&2&3&\infty&\infty&\infty\\1&0&1&2&\infty&1&\infty\\2&1&0&1&1&\infty&\infty\\3&2&1&0&\infty&\infty&\infty\\\infty&\infty&1&\infty&0&1&2\\\infty&1&\infty&\infty&1&0&1\\\infty&\infty&\infty&\infty&2&1&0\end{bmatrix},$$

$$W_1=\begin{bmatrix}\infty&\infty&\infty&\infty&3&2&\infty\\\infty&\infty&\infty&\infty&2&\infty&2\\\infty&\infty&\infty&\infty&\infty&2&3\\\infty&\infty&\infty&\infty&2&3&\infty\\3&2&\infty&2&\infty&\infty&\infty\\2&\infty&2&3&\infty&\infty&\infty\\\infty&2&3&\infty&\infty&\infty&\infty\end{bmatrix},$$

$$W_2=\begin{bmatrix}\infty&\infty&\infty&\infty&\infty&\infty&3\\\infty&\infty&\infty&\infty&\infty&\infty&\infty\\\infty&\infty&\infty&\infty&\infty&\infty&\infty\\\infty&\infty&\infty&\infty&\infty&\infty&4\\\infty&\infty&\infty&\infty&\infty&\infty&\infty\\\infty&\infty&\infty&\infty&\infty&\infty&\infty\\3&\infty&\infty&4&\infty&\infty&\infty\end{bmatrix},$$

$$W=\begin{bmatrix}0&1&2&3&3&2&3\\1&0&1&2&2&1&2\\2&1&0&1&1&2&3\\3&2&1&0&2&3&4\\3&2&1&2&0&1&2\\2&1&2&3&1&0&1\\3&2&3&4&2&1&0\end{bmatrix}.$$

4 实例分析

利用 Google 地图获取成都市 1~10 路公交车线路及其 214 个车站的经纬度。为便于应用上述算法，将每个车站名分别映射为 1~214 的数字编号，利用 Matlab 2009a 建立这 10 条线路组成的换乘网络 Space P 的可达矩阵；根据第 2 章的算法获得网络各连边权重，建立加权距离矩阵 W_0 ；利用第 3 章的算法，计算最小乘车次数矩阵 D 和最短路程矩阵 W ；分别计算矩阵 D 和 W 中元素的平均值，得到网络平均最短路径、平均乘车次数及平均换乘次数(换乘次数=乘车次数-1)。然后利用 Floyd 算法直接计算矩阵 W_0 中各车站对间的距离，得到平均最短路径长度及其对应的乘车次数及换乘次数。两种算法的结果对比见表 1，由本文算法得到平均最短路径长度增加了 3.69 km，但平均换乘次数下降了 0.64 次，更符合乘客的出行习惯。

随机选择一些车站进行最优换乘路径选取试验，并与 Floyd 算法计算出的最短路径和换乘次数进行比较。从成都东客站东广场(车站编号 40)到红星路一段(车站编号 143)，利用本文算法得出的最短路径长度为 8.94 km，具体换乘方案为：在成都东客站东广场乘坐 2 路至二环路东三段(车站编号 35)，换乘 5 路至红星路口(车站编号 91)，再换

乘 7 路至红星路一段，总换乘次数为 2 次。利用 Floyd 算法计算加权距离网络的最短路径，得到的方案为：在成都东客站东广场乘坐 2 路至双桥子(车站编号 37)，换乘 10 路至水碾河(车站编号 58)，继续换乘 3 路至玉沙路(车站编号 51)，最后换乘 7 路至红星路一段；该方案总距离为 8.10 km，换乘次数为 3 次。

分析上述结果不难发现，在绝对行驶路径长度增长不大的情况下，第一种方案较第二种方案减少了 1 次换乘，说明利用本文算法得到的方案在保证换乘次数最少的基础上，得到的路径也基本最短，证明了算法的有效性。

5 结论

本文基于公交复杂网络中的换乘网络 Space P 的拓扑结构，以站间距离为权重，利用车站的经纬度计算站间边权，建立了距离加权公交换乘复杂网络模型，提出了一种在保证站间换乘次数最小的情况下行驶路径也相对最短的公交换乘算法。该算法克服了普通最短路算法不能直接计算加权公交复杂网络最短路的局限性。但是，由于利用经纬度确立的站间距离为车站间的直线距离，与实际情况有一定误差，因此，利用本文算法无法得出精确的结果。同时，文中建立算法的基础是假设乘客都倾向于选择换乘次数少的乘车方案，与现实情况也有一定差距。未来可以从减小站间距离误差、提出更合理的公交出行路径选择模型入手，进一步深化研究。

(下转第 85 页)

表 1 Floyd 算法与本文算法对比

Tab.1 Comparison between the Floyd algorithm and the proposed algorithm

算法	平均最短路径/km	平均乘车次数	平均换乘次数
Floyd	11.14	2.73	1.73
本文算法	14.83	2.09	1.09