公共交通出行线路最优选择

连新泽

（温州大学数学与信息科学学院，浙江温州　325035）

摘　要：运用数学理论和方法以及计算机代数系统 Maple，建立了公共交通（包括公汽、地铁等）出行线路选择问题的算法．针对 2007 年全国大学生数学建模竞赛 B 题，对任意给定起始站点和目的站点以及乘客的选择模式，应用该算法得到了乘客需求的最佳换乘公交路线及所花的时间和费用．

关键词：最短路径；最少换乘；公共交通系统

中图分类号：O141.4　　**文献标识码**：A　　**文章编号**：1006-0375(2008)03-0037-07

公共交通（包括公汽、地铁等）是现代化城市的动脉，以其经济、安全、相对环保等特点，成为人们出行的首选．随着城市化进程的加快，城市的公共交通系统有了很大改观，大型城市的公交线路已达800多条．如此众多的公交线路使得人们出行更加通畅、便利，但人们同时也面临多条线路的选择问题．因此，提供方便、快捷、经济、高效的公共交通出行线路选择方案，不仅可以方便人们的出行和生活，同时也有利于提高城市的交通运输效率，展示现代城市信息化的风貌．

1959年，Dijkstra[1]提出了最短路径算法。该算法由于适应网络拓扑变化的稳定性，在计算机网络拓扑路径选择以及地理信息系统（GIS）中得到了广泛应用[2-3]．但是Dijkstra算法是针对从某点到其余各点的最短路径，对大型数据的计算所需时间较长，所以Dijkstra算法并不适合公交最短路径查询[4,5]．随着对公交线路选择问题研究的不断深入，当前流行的地理信息系统（GIS）借助其空间分析能力和可视化表达发挥着重要作用．马良河[6]研究了城市公交线路网络图的最短路径与乘车路线问题，Choi K[7]利用GIS技术讨论了从街道地理数据产生公交线路和站点的问题，黄正东[8]研究了GIS中公交实体与基础路网的关联。针对现有的城市公交运行线路、站点分布和城市的道路网络特点，GIS技术对于建立城市公交查询系统有极大帮助。但由于其计算量巨大，只能较好地应用于小城市的公交系统。在大城市中使用GIS尚不能达到理想的速度和结果．

事实上，人们出行的公交线路主要基于两种方式：单纯乘坐公汽或地铁；乘坐公汽和地铁．考虑线路的选择、主要乘车时间、换成次数及费用等因素，并根据相关心理学研究结果，换乘3次是人们的心理承受上限，本文建立了公交出行线路选择模型及相应的优化算法．

1 模型假设与符号说明

1.1 模型假设

（I）假设公交到达最后一站，乘客必须下车；

收稿日期：2007-11-10

基金项目：大学生数学建模竞赛研究和实践（314040310407）

作者简介：连新泽（1980- ），男，浙江温州人，助理实验师，学士，研究方向：计算机数学

（II）假设环行路线是双向环行；

（III）假设同一地铁站对应的任意两个公共汽车站直接可以通过地铁换乘（无需支付地铁费）；

（IV）假设换乘的次数不超过两次.

1.2 符号说明

L：所有的公交路线；

L_i：i 为奇数表示第[i/2][①]条公交路线的正方向，i 为偶数表示第 $i/2$ 条公交路线的反方向；

S_j：经过站点 j 的所有公交路线集合；

T_0：可步行的最大时间；

Z_a：与站点 a 的步行时间不大于 T_0 的所有站点集合（包含站点 a 本身）.

2 模型与分析

首先考虑单纯公汽交通的情况，然后再考虑公汽线路与地铁线路的情况，最后考虑增加站点与站点之间的步行时间的因素．为了便于描述，以下在时间费用优先（即时间优先）的模式下进行讨论.

2.1 公共交通系统中只有公汽

由前面假设（I）和假设（II），把同一路公共汽车的交通线路分为方向相反的两路线路，任意两个公汽站点 a，b，其中 a 为起始站点，b 为目的站点，经过 a，b 站点的所有公汽路线集合记为 S_a，S_b．由前面假设（IV），起始站点 a 到目的站点 b 有三种换乘方式：直达（不需要换乘），换乘一次和换乘两次.

（1）直达：乘坐 L_i 可以从站点 a 直达站点 b 满足如下条件：

（i）$L_i \in S_a \cap S_b$；

（ii）在公汽路线 L_i 中，站点 a 在站点 b 的前面.

（2）换乘一次：起始站点 a 先乘 L_i 再换乘 L_j 到目的站点 b 满足如下条件：

（i）$\exists c, c \in L_i \cap L_j$；

（ii）在公汽路线 L_i 中，站点 a 在 c 的前面．在公汽路线 L_j 中，站点 c 在 b 的前面．其中 $L_i \in S_a$，$L_j \in S_b$.

（3）换乘两次：起始站点 a 先乘 L_i，换乘 L_k，再换乘 L_j 到目的站点 b 满足如下条件：

（i）$L_k \in L$，$\exists c, d$，其中 $c \in L_k \cap L_i$，$d \in L_k \cap L_j$；

（ii）在公汽路线 L_i 中，站点 a 在 c 的前面，在公汽路线 L_k 中，站点 c 在 d 的前面，在公汽路线 L_j 中，站点 c 在 b 的前面，其中 $L_i \in S_a$，$L_j \in S_b$，$L_k \in L$.

根据三种换乘方式的定义，设计函数 ind（fare, L, nline, node1, node2, M），该函数用来判断乘坐 L[nline]路公汽能否从站点 node1 到达站点 node2，如果能够到达，返回站点 node1 和 node2 在公汽线路 L[nline]上距离最短的两个位置（公汽路线可能多次经过同一站点）、时间、费用，否

① [n]表示对 n 上取整.

则返回 false.

算法 1：ind 算法

Input fare,L,nline,node1,node2

output re=[时间,费用],[位置,位置]

step1 计算 K1；#K1 为 node1 在线路 L[nline]中的所有位置集合；

step2 计算 K2；#K2 为 node2 在线路 L[nline]中的所有下标集合；

step3 重复下列步骤直到遍历所有的[k1[i],k2[j]]；#k1[i]∈K1,k2[j]∈K2；

step3.1 如果 k2[j]>k1[i] and k2[j]-k1[i]<m，则 m=k2[j]-k1[i]，把[k1[i],k2[j]]的时间、费用记录在 re 变量中，否则重复 step3；

step4 返回 re.

根据前述直达、换乘的定义，在算法 ind 的基础上建立以下三个不同换乘次数的最优算法：

算法 2：直达最优算法 node12set[0] (fare,L,a,b)

input fare,L,node1,node2

output re=[路线,[时间,费用],[位置,位置]]

step1 s1=pnode(L,node1)；#计算经过 node1 的所有线路；

step2 s2=pnode(L,node2)；

step3 s=s1 intersect s2；

step4 m=F=infinity；

step5 i 从 1 到 nops(s)；#nops(s)表示集合 s 中的元素个数

step5.1 temp=ind(fare,L,s[i],node1,node2)；

step5.2 如果 temp<>false and temp[1,1]<m，则 m=temp[1,1]，F=temp[1,2]，re=[[s[i],temp]]；

step5.3 如果 temp<>false and temp[1,1]=m and temp[1,2]<F，则 F=temp[1,2]，re=[[s[i], temp]]；

step5.4 重复 step5 直到 i>nops(s)；

step6 返回 re.

算法 3：换乘一次最优算法 node12set[1] (fare,L,a,b)

input fare,L,node1,node2

output re=[路线,[时间,费用],[位置,位置]],[路线,[时间,费用],[位置,位置]]

step1 s1=pnode(L,node1)；s2=pnode(L,node2)；

step2 m=F=infinity；

step3 i 从 1 到 nops(s1)；j 从 1 到 nops(s2)；

step3.1 s=jd(L[s1[i]],L[s2[j]])；#求出公汽线路 L[s1[i]],L[s2[j]]的交点；

step3.2 k 从 1 到 nops(s)；

step3.2.1 re1:=ind(fare,L,s1[i],node1,s[k])；re2:=ind(fare,L,s2[j],s[k],node2)；

step3.2.2 如果站点 node1 可以直达到站点 s[k]，并且站点 s[k]也可以直达到站点 node2，则执行 step3.2.3，否则重复 step3.2；

step3.2.3 计算站点 s1[i]到 s2[j]的时间和费用，分别赋值给 temp1 和 temp2；

step3.2.4 如果 temp1 的时间少于 m，则 m=temp1；F=temp2；re=[[s1[i],re1],[s2[j],re2]]；

否则，如果时间等于 m，并且费用小于 F，则 F=temp2；

re=[[s1[i],re1],[s2[j],re2]]；

step3.2.5 重复 step3.2 直到 k>nops(s)；

step3.3 重复 step3 直到 i>nops(s1)；j>nops(s2)；

step4 返回 re.

算法 4：换乘两次最优算法 node12set[2] (fare,L,a,b)

input fare,L,node1,node2

output re=[路线,[时间,费用],[位置,位置]],[路线,[时间,费用],[位置,位置]],[路线,[时间,费用],[位置,位置]]

step1 计算 s1=pnode(L,node1)；s2=pnode(L,node2)；

step2 s=s1 intersect s2；s1=s1 minus s；s2= minus s；

step3 m=F=infinity；

step4 i 从 1 到 nops(s1)；j 从 1 到 nops(s2)；k 从 1 到 nops(L)；

step4.1 如果 k<>s1[i]并且 k<>s2[j]，则把第 k 与第 s1[i]路交通路线的公共站存在变量 snodeik 中，把第 k 与第 s2[j]路交通路线的公共站存在变量 snodejk 中，否则重复 step4；

step4.2 k1 从 1 到 snodeik 个数；k2 从 1 到 snodejk 个数；

step4.2.1 re1:=ind(fare,L,s1[i],node1,snodeik[k1])；

step4.2.2 re2:=ind(fare,L,k,snodeik[k1],snodejk[k2])；

step4.2.3 re3:=ind(fare,L,s2[j],snodejk[k2],node2)；

step4.2.4 如果 re1,re2,re3 都不为 false，则求出 node1 到 node2 的时间和费用，分别记为 temp1 和 temp2，否则重复 step4.2；

step4.2.5 如果时间 temp1 小于 m，则 m=temp1；F=temp2；node1 到 node2 的选择路线及时间费用信息保存在 re 中；否则若果时间 temp1 等于 m 并且费用 temp2 小于 F；则 F＝temp2；node1 到 node2 的选择路线及时间费用信息保存在 re 中；

step4.2.6 重复 step4.2 直到 k1>nops(snodeik)；k2>nops(snodejk)；

step4.3 重复 step4；直到 i>nops(s1)；j>nops(s2)；k>nops(L)；

step5 return re.

利用算法 2、算法 3、算法 4 分别计算相应的解，再从这些解中挑选出最优解，即为单纯公汽时的最优出行选择方案.

2.2 公共交通系统中含有公汽和地铁

考虑到地铁转乘比较方便，所以假定地铁转地铁不计入换乘次数. 据此直达的出行路线有两种：地铁或公汽直达；换乘一次的出行线路三种：公汽-地铁、公汽-公汽（2.1 已讨论）、地铁-公汽；换乘两次的出行线路四种：公汽-公汽-公汽（2.1 已讨论）、公汽-公汽-地铁、公汽-地铁-公汽、地铁-公汽-公汽，下面简述这几种换乘情况下的最优解算法.

直达最优算法：利用算法 2 计算公汽直达解，地铁直达解，再进行比较，返回最优解；

换乘一次最优算法：利用算法 3 计算公汽-地铁、公汽-公汽（2.1 已讨论）、地铁-公汽换乘

一次的最优解，再进行比较，返回最优解；

换乘两次最优算法：利用算法 4 计算公汽–公汽–公汽（2.1 已讨论）、公汽–公汽–地铁、公汽–地铁–公汽、地铁–公汽–公汽换乘两次的最优解，再进行比较，返回最优解；

利用以上三种算法分别计算最优解，比较后返回最优解的结果即为公汽–地铁情形下最优出行选择方案.

2.3 公共交通系统考虑站点间步行时间的模型

假设已知任意两站点 a,b 之间的步行时间以及可步行的时间 T_0（即一般人不愿走很长的路），根据 T_0 就可以求得站点 a，b 的可步行站点集合 Z_a, Z_b. 在这种情况下，直达、转乘一次、转乘两次的情况分别为：

直达：如果有公交路线先经过集合 Z_a 中的站点，又经过集合 Z_b 中的站点，乘坐该路线可以直达.

转乘一次：如果有路线 L_1 经过集合 Z_a 中的站点 a_0，L_2 经过 Z_b 中的站点 b_0，并且路线 L_1，L_2 中有步行时间少于 T_0 的站点 a_1，b_1（$a_1 \in L_1$，$b_1 \in L_2$），且 L_1 是后经过 a_1，L_2 是先经过 b_1，则从站点 a 步行出发到 a_0，乘坐 L_1 路公交至 a_1，下车后步行到 b_1 再转乘 L_2 路公交至 b_0，下车后步行到 b，该线路为换乘一次线路.

转乘两次：如果有路线 L_1 经过集合 Z_a 中的站点 a_0，L_3 经过 Z_b 中的站点 b_0，存在 L_2 上站点 c_1，c_2（c_1 在 c_2 前面）且站点 c_1 与线路 L_1 上的站点 a_1（a_1 在 a_0 后面）步行时间少于 T_0，站点 c_2 与 L_2 上的站点 b_1（b_1 在 b_0 前面）的步行时间少于 T_0，则从站点 a 步行出发到 a_0，乘坐 L_1 路公交至 a_1，下车后步行到 c_1，再转乘 L_2 路公交至 c_2，下车后步行到 b_1 再转乘 L_3 路公交至 b_0，下车后步行到 b，该线路为换乘二次线路.

用同样的方法求出直达、换乘一次、换乘两次的解，再从这些结果中挑选出最优解，即为最优出行选择方案.

3 应用实例

3.1 公共交通系统中只含公汽线路

利用 2.1 建立的算法，对 2007 年全国大学生数学建模竞赛 B 题相关数据①进行计算，得到如下结果。见表 1.

3.2 公共交通系统中含公汽与地铁线路

在公共交通系统中含公汽与地铁线路的情形下，单考虑时间优先约束下的最优选择情况，见表 2.

4 结　论

本文根据乘客的需求，把公共交通线路选择问题分为单目标和二次规划的最优化问题，先求出换乘 i（i=0,1,2）次的最优路线，再比较这三种换乘下返回的结果，从而找出最优路线. 本文给出的算法思路清晰，互动性高，可为公交线路查询系统的设计提供参考.

---

① 全国大学生数学建模竞赛 2007 年赛题[EB/OL]. [2007-09-30]. http://mcm.edu.cn/mcm07/problems2007c.asp.

表 1 公汽情形下最优出行选择结果

模式	起始站→目的站	最佳线路[①]	时间/min	费用/元
时间优先 M=1	S3359→S1828	[L15,[3359,2903]],[L 485,[2903,1784]],[L 167,[1884,1828]]	64	3
	S1557→S0481	[L84,[1557,1919]],[189,[1919,3186]],[460,[3186,481]]	106	3
	S0971→S0485	[L13,[971,2517]],[290,[2517,2519]],[469,[2159,485]]	103	3
	S0008→S0073	[L 43,[8,1383]],[290,[1383,2184]],[345,[2184,73]]	67	3
	S0148→S0485	[L308,[148,36]],[156,[36,2210]],[417,[2210,485]]	106	3
	S0087→S3676	[L21,[87,88]],[L231,[88,427]],[L97,[427,3676]]	46	3
费用优先 M=2	S3359→S1828	[L15,[3359,2903]],[L485,[2903,1784]],[L167,[1884,1828]]	64	3
	S1557→S0481	[L84,[1557,1919]],[L189,[1919,3186]],[L460,[3186,481]]	106	3
	S0971→S0485	[L13,[971,2517]],[L290,[2517,2519]],[L469,[2159,485]]	103	3
	S0008→S0073	[L159,[8,291]],[L58,[291,73]]	83	2
	S0148→S0485	[L308,[148,36]],[L156,[36,2210]],[L417,[2210,485]]	106	3
	S0087→S3676	[L454,[87,3496]],[L209,[3496,427]]	65	2
默认模式 M=3	S3359→S1828	[L436,[3359,1784]],[L167,[1784,1828]]	101	3
	S1557→S0481	[L84,[1557,1919]],[L189,[1919,3186]],[L460,[3186,481]]	106	3
	S0971→S0485	[L13,[971,2184],[L417,[2184,485]]	128	3
	S0008→S0073	[L159,[8,291]],[L58,[291,73]]	83	2
	S0148→S0485	[L308,[148,36]],[L156,[36,2210]],[L417,[2210,485]]	106	3
	S0087→S3676	[L454,[87,3496]],[L209,[3496,427]]	65	2

表 2 公汽－地铁情形下最优出行选择结果

模式	起始站→目的站	最佳线路[②]	时间/min	费用/元
时间优先 M=1	S3359→S1828	[L15,[3359,2903]],[L485,[2903,1784]],[L167,[1884,1828]]	64	3
	S1557→S0481	[L84,[1557,1919]],[L189,[1919,3186]],[L460,[3186,481]]	106	3
	S0971→S0485	[L13,[971,2517]],[L290,[2517,2519]],[L469,[2159,485]]	103	3
	S0008→S0073	[L200,[8,2534]],[T1,[15,12]]], [T2,[12,25]]], [L103,[525,73]]	53.5	5
	S0148→S0485	[L308,[148,36]],[L156,[36,2210]],[L417,[2210,485]]	106	3
	S0087→S3676	[L21,[87,630]],[T2,[29,36]],[L97,[427,3676]]	39	5

参考文献

[1] Dijkstra E W. A note on two problems in connection with graphs [J]. Numerische Mathematik, 1959, 1(1): 269-271.

[2] 古凌岚. GIS 最短路径分析中 Dijkstra 算法的优化[J]. 计算机与数字工程, 2006, 34(12): 53-56.

[3] 张雪燕, 黄寅, 杨晟刚. 一种改进的 Dijkstra 算法应用于嵌入式 GIS 系统[J]. 计算机工程与设计, 2007, 28(2): 412-414.

[4] 梁虹, 袁小群, 刘蕊. 一种新的公共交通数据模型与公交查询系统实现[J]. 计算机工程与应用, 2007, 43(3): 234-238.

① 最佳路线的数据[Li, [n1, m1]], [Lj, [n2, m2]]表示乘第 i 路公汽, 在站点 n1 上车一直乘到 m1 站下车, 再转乘第 j 路公汽, 在站点 n2 上车一直乘到站点 m2 下车.

② Ti 表示 i 号地铁线路.

[5] 王莉，李文权．公共交通最佳路径算法[J]．东南大学学报, 2004, 34(2): 264-267.

[6] 马良河．城市公交线路网络图的最短路与乘车路线问题[J]．数学的实践与认识, 2004, 34(6): 38-44.

[7] Choi K, Jang W. Development of a transit network form a street map database with spatial analysis and dynamic segmentation [J]. Transportation Research: Part C, 2000, 8: 129-146.

[8] 黄正东．公交实体的详细表达及其在出行系统中的应用[J]．武汉大学学报, 2003, 36(3): 69-75.

Optimum Route Choice on Public Traffic System

LIAN Xinze

(School of Mathematics and Information Science, Wenzhou University, Wenzhou, China 325035)

Abstract: In this paper, some algorithms about optimum route choice on public traffic system are established by using computer algebra system Maple and mathematical theories. Based on these algorithms, this paper successfully solves the B problem of the China Undergraduate Mathematical Contest in Modeling 2007.

Key words: Optimum route; Least transfer; Public traffic system

（编辑：王一芳）