

基于客流量预测的铁路分段购票公平性研究

摘要

由于我国人口流动性大，且相对于其他运输工具，火车既安全又便宜，所以火车成为绝大多数人的出行首选。然而，购票难，特别是节假日期间的一票难求，一直是人们热议的话题。我国铁道总局一直实行“同车禁止分段购票”的规则，但是为何禁止分段购票，是否应该放开分段购票，却没有统一、服众的说法。

在此背景下，本文将研究两个问题：问题一，定量分析分段购票是否会影响乘客购票的公平性。问题二，讨论是否存在允许分段购票，既可以保证公平，又可以提高运力的情况，并进一步讨论这些规则对客运运力资源调配系统复杂度和效率的影响。

针对问题一，定量分析分段购票是否影响购票公平性，本文分三个步骤进行。第一，建立公平性指标，用允许分段总出行客流量与禁止分段总出行客流量之差，来衡量分段购票是否影响乘客购票公平性。第二，在有限站点的铁路上划分出行方案，依据终点、区间、分段数逐层建立起允许分段条件下的可列个出行方案。第三，按照一定顺序遍历所有出行方案，计算成功出行总人数。从而建立起开售时即允许分段的售票模型。通过软件求解，我们发现，分段数目影响分段出行总人数结果：在分段数为 2 或 3 时总出行客流量有所提高，而分段数过大时，总客流量降低。

针对问题二，讨论是否存在允许分段，既保证公平又提高运力的情况，并进一步讨论该规则对客运系统复杂度、效率的影响。根据问题一的模型与求解，我们提出两种符合要求的情况：1.直达先购票，余票再分段。2. 限制分段数目。实务中我们无法得知购买直达票的乘客是否已经全部购完，但是可用时间刻画，规定在开车前几天才启动分段售票，此时大部分直达购票乘客应已经完成购买。综合模型和实例分析，我们认为分段为 2 或 3 时较为合理，可将公平与效率兼顾。进一步，根据以上情况，我们从算法和实务两方面讨论复杂度和效率，虽然分段售票一定程度提高复杂度、降低效率，但是这是在可以接受范围内的。

最后，需要结合上述讨论，给铁路客运相关部门写一份公开信，谈谈对“禁止分段购票”规则的看法和建议。本文依照上述分析，简要给出了看法、建议：我们认为分段售票的时间以及分段数选择很重要，如果在起售时即允许分段售票，或允许分段的数目过大，可能影响购票公平性。但是可以在售票末期开放分段售票，或者限制分段数，从而提高铁路客运运力，同时也不影响公平性。

关键词：分段购票 铁路旅客运输 票额分配 铁路客运运力

目录

摘要	2
一、问题提出	4
二、问题分析	4
三、模型假设	4
四、符号说明	5
五、模型建立与求解	6
5.1 问题一	6
5.1.1 所需数据	6
5.1.2 公平性以及运力评价指标	6
5.1.3 分段售票模型	7
5.1.4 分段数目影响分析	10
5.1.5 实例分析	10
5.2 问题二	12
5.2.1 公平且提高运力的情况分析	12
5.2.2 客运运力资源调配系统复杂度和效率分析	13
5.3 公开信	13
六、模型推广	14
七、模型评价	14
7.1 模型优点	14
7.2 模型缺点	14
7.3 模型优化	14
八、参考文献	14
附录	15

一、问题提出

中国是个人口大国，且人口分布与经济发展不均衡，因此人口流动性较大。相对于其他运输工具而言，乘坐火车既安全又便宜，还能准时准点的开运和到站，所以成为绝大多数人的出行首选。然而，购票难，特别是节假日期间的一票难求，一直是人们热议的话题。我国铁道总局一直实行“同车禁止分段购票”的规则，但是为何禁止分段购票，却没有强有力的理由。

2019年2月18日，一则“不满‘同车禁止分段购票’律师将‘铁总’告上法庭”的消息引发网友热议，一时分段购票引起人们的广泛关注。

2019年2月22日《广西日报》第006版发表了一篇题为“赞 禁止分段购票符合公平要义”的评论，表示同车禁止分段购票有其公平合理性，但同时也应动态的灵活调整票额。

关于分段购票，为何禁止分段购票，是否应该放开分段购票，是有待解决而研究甚少的问题。

二、问题分析

问题一，定量分析分段购票是否会影响乘客购票的公平性。第一步，建立公平性、运力评价指标：若分段购票能使更多乘客成功出行，则不影响购票公平；不然，则影响公平。第二步，分别计算禁止、允许分段购票情况下能成功出行的总人数，并进行比较。因为现行规定是禁止同车分段购票，所以由查得的实际票额售出统计表，可直接计算出禁止分段购票情况下成功出行的总人数。然后，通过区间票额、预测客流量，求得若起售时就允许分段购票能成功出行的总人数，同时讨论了分段数目对结果的影响。

问题二，讨论是否存在允许分段购票，既可以保证公平，又可以提高运力的情况，并进一步讨论这些规则对客运运力资源调配系统复杂度和效率的影响。根据问题一中的求解，我们提出两种情况：1、先让所有能购买直达票出行的乘客购票，再把余票分段售票。这种情况显然既保证了公平又提高了运力，虽然实际上我们无法得知购买直达票的乘客是否已经全部购完，但是可用时间刻画，规定在开车前几天才启动分段售票。2、限制分段数目。由问题一中的分析，可知分段数目对成功出行总人数有影响，分段太细会减少总出行人数，分段为3时较为合理。进一步根据以上情况，从算法和实务两方面讨论复杂度和效率。

最后，需要结合上述讨论，给铁路客运相关部门写一份公开信，谈谈对“禁止分段购票”规则的看法和建议。本文依照上述分析，给出了看法、建议。

需要注意，铁道交通领域票额指票的数量，而非面值，本文中该词均是此意。

三、模型假设

- 1.固定区间票额固定，不进行动态调度。
- 2.所有席位视为一致，不区分座卧铺、一二等座等。
- 3.旅客在已知全部余票信息的情况下进行购票。

四、符号说明

符号	说明
$0, 1, \dots, H, \dots, K, \dots, L$	列车单程沿途站点
P	预测客流量矩阵
K	实际客流量矩阵
C	票额分配矩阵
p_{ij}	任意两个站点间的预测客流量
k_{ij}	任意两个站点间的实际客流量
c_{ij}	任意两个站点间的票额
SP	分段售票条件下成功出行的客流矩阵
sp_{ij}	分段售票条件下任意两站间成功出行客流量
a	公平性指标
b	运力指标
N	列车座位数

五、模型建立与求解

5.1 问题一

5.1.1 所需数据

某次列车单程沿途站点示意图如图 1，分别记各站点为 $0, 1, \dots, L$ 。

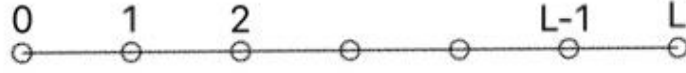


图 1 列车沿途站点示意图

建立三个矩阵，预测客流矩阵 P ，实际客流矩阵 K ，票额分配矩阵 C 。分别记为

$$P = (p_{ij})_{(L+1) \times (L+1)}, \quad K = (k_{ij})_{(L+1) \times (L+1)}, \quad C = (c_{ij})_{(L+1) \times (L+1)}$$

其中， p_{ij} 表示任意两个站点间的预测客流量， k_{ij} 表示任意两个站点间的实际客流量， c_{ij} 表示任意两个站点间的票额。其中 i 为出发站， j 为出行站（ $i=0, 1, \dots, L-1$ ， $j=i+1, i+2, \dots, L$ 其他元素记为 0）。

P 矩阵可由客流量预测模型^[4]求得， K 矩阵由真实数据可得。因为现行规定禁止同车分段购票，为了保证长途旅客出行，铁路部门采取区间限售的方法，即长途分配更多票额，短途票额较少。而在本文模型中，拟分析起售时就允许分段售票对成功出行总人数的影响，因此不能采用现行票额分配方案。本文按照该车次实际客流占预测客流的比重，依照各站点间预测客流比例分配票额，形成 C 矩阵。

三个矩阵的特征：都是下三角阵，对角线上元素都为 0。

5.1.2 公平性以及运力评价指标

记 ΣK 为实际客流总量。

$$\Sigma K = \sum_{i,j} k_{ij}$$

建立允许分段售票条件下成功出行的客流矩阵 SP ， $SP = (sp_{ij})_{(L+1) \times (L+1)}$ 其中 sp_{ij} 表示任意两个站点间成功出行客流量，通过后文分段售票分配模型，可最终求得成功出行总人数，记为 ΣSP 。

$$\Sigma SP = \sum_{i,j} sp_{ij}$$

定义公平性评价指标 a

$$a = \Sigma SP - \Sigma K$$

$a > 0$ ，表示允许分段规则下成功出行总人数提高，分段不影响公平； $a < 0$ ，

表示允许分段规则下成功出行总人数减少，分段影响公平。

把客座率^[1] b 作为评价运力的指标，

$$b = b_2 - b_1$$

$$b_1 = \frac{\sum K}{N}, b_2 = \frac{\sum SP}{N}$$

其中， N 为该列车座位数。 $b > 0$ ，表示分段售票提高座位利用率，提高铁路客运运力； $b < 0$ ，则表示降低运力。

5.1.3 分段售票模型

1、分段售票模型

在起售时即允许分段售票的条件下，本文划分出可数可列个出行方案，即某人从 i 点出发到达 j 点的购票方案，划分依据依次递进，它们分别是终点、区间、分段数，从而计算出分段情况下的出行总人数 $\sum SP$ 。

简单地说，模型以 P 、 C 、 SP 矩阵为基础，把出行方案作为变量，依次遍历所有出行方案，过程中不断更新 P 、 C 、 SP 矩阵，最后得到最终结果。图 2 是这一过程的流程图。

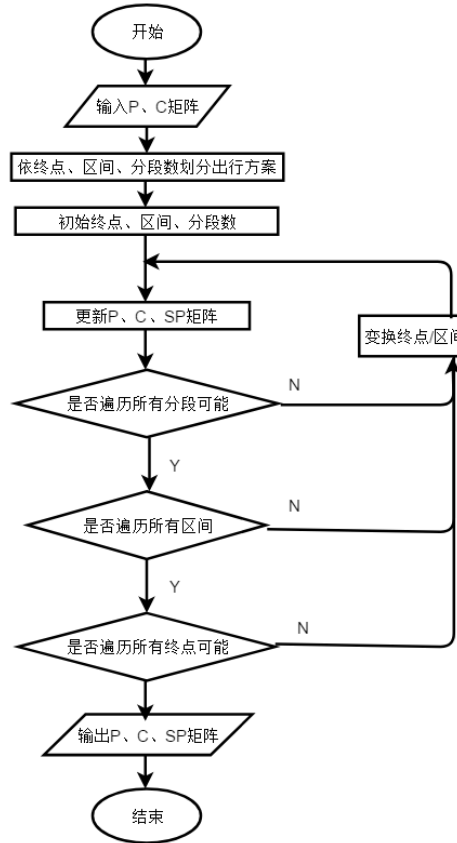


图 2 问题一流程图

第一步，依据终点把出行方案划分为以 L 为终点，以 $L-1$ 为终点，……，以 1 为终点，共 L 种可能。

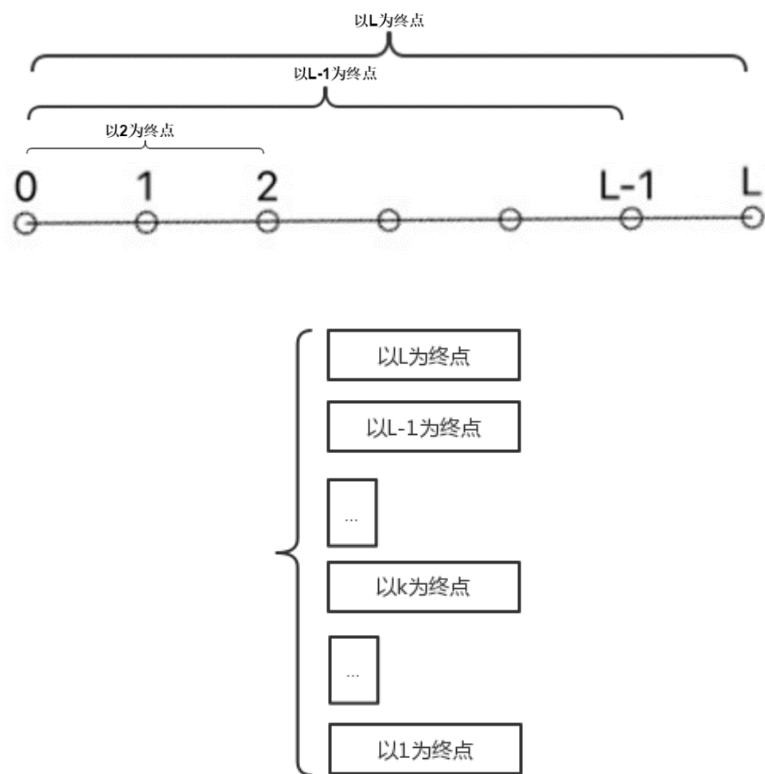


图 3 依终点划分

第二步，对于某个以 K 为终点的选项($K=L, L-1, \dots, 1$)，改变起点 H 的位置 ($H=0, 1, \dots, K-1$)，对于任意 $H \geq 1$ ， H 将会把 $[0, K]$ 区间划分为 $[0, H]$, $[H, K]$ 两个区间，称 $[0, H]$ 为左区间， $[H, K]$ 为右区间。变化 H ，会有如下可能，共 K 种：

$[0, K], \{[1, K], [0, 1]\}, \{[2, K], [0, 2]\}, \dots, \{[K-1, K], [0, K-1]\}$

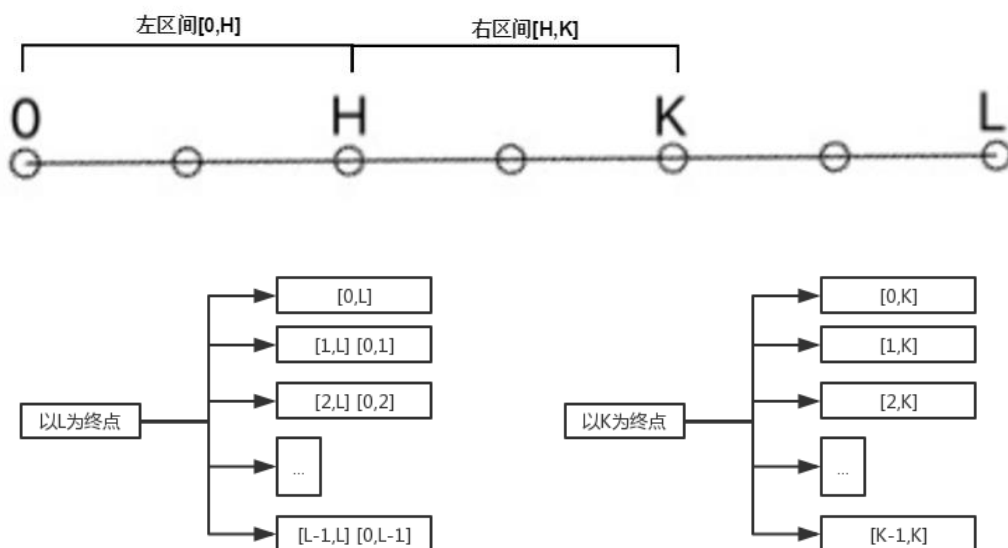


图 4 依区间划分

其中，当 $K=L$ 时，会依次讨论完 $[0, 1], [0, 2], \dots, [0, L-1]$ ，所以 $K < L$ 时不再讨论这些区间。

第三步，对于某个 $[0,H]$ 区间，依据分段数可划分为 1 段（直达）、2 段、...、 H 段，共 H 中划分可能。若分 i 段，则有 C_{H-1}^i 中分段方式。实际上左右区间相互独立，且研究方法一致，因此对 $[H,K]$ 不再赘述。

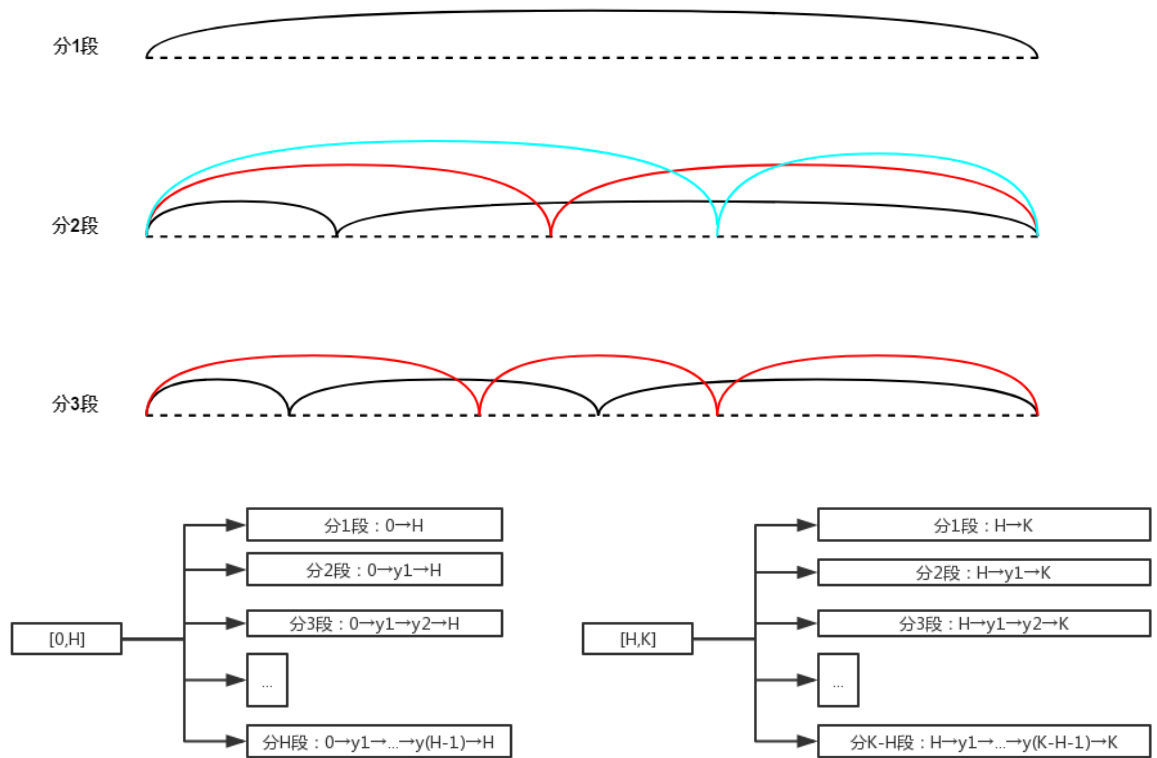


图 5 依分段数划分

最后，按照划分的方案依次分配票额。具体做法如下：

A. 以 L 为终点

a. 区间 $[0,L]$

(1)分 1 段（直达），(2)分 2 段，③分 3 段，...，(L)分 L 段

b. 区间 $[1,L],[0,1]$

$[1,L]$: (1)分 1 段（直达），(2)分 2 段，③分 3 段，...，(L-1)分 $L-1$ 段

$[0,1]$: 分 1 段

c. 区间 $[2,L],[0,2]$

$[2,L]$: (1)分 1 段（直达），(2)分 2 段，③分 3 段，...，(L-2)分 $L-2$ 段

$[0,2]$: (1)分 1 段（直达），(2)分 2 段

...

l. 区间 $[L-1,L],[0,L-1]$

$[L-1,L]$: 分 1 段

$[0,L-1]$: (1)分 1 段（直达），(2)分 2 段，③分 3 段，...，(L-1)分 $L-1$ 段

B. 以 $L-1$ 为起点

（中间过程同上）

...

L.以 1 为起点

以[0,L]区间为例，对于 A.a.(1)，记 $t=\min\{p_{0L}, c_{0L}\}$ ，则 t 为[0,L]区间能通过直达方式出行的人数。此时，

$$p_{0L} = p_{0L} - t$$

$$c_{0L} = c_{0L} - t$$

$$sp_{0L} = sp_{0L} + t$$

对于 A.a.(2)，若选取分段中间点 I ，记 $t=\min\{p_{0L}, c_{0I}, c_{IL}\}$ ，则 t 为[0,L]区间能通过分段方式 $0 \rightarrow I \rightarrow L$ 出行的人数。

$$p_{0L} = p_{0L} - t$$

$$c_{0I} = c_{0I} - t$$

$$c_{IL} = c_{IL} - t$$

$$sp_{0L} = sp_{0L} + t$$

每做一步，更新 P、C、SP 矩阵，最终得到成功出行总人数 $\sum SP$ 。

2、分段售票原则

分段售票的目的是在最大程度保证乘客的出行需求的同时，使列车运力得到充分利用，因此遵循以下三个原则进行分配。

(1)先长途后短途。

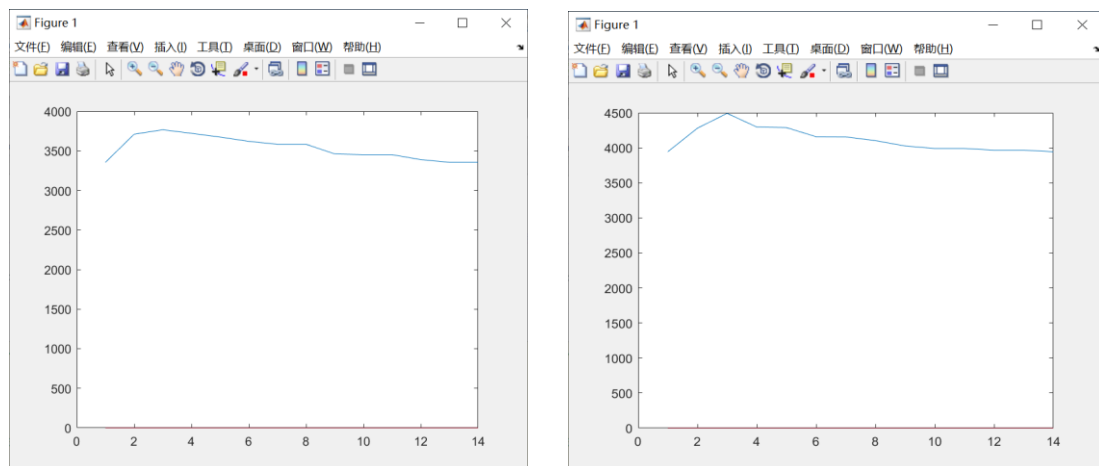
(2)先直达后分段。

(3)左右区间相互独立，左区间随右区间一同研究。

三个原则有现实合理性，且上述分段售票模型是符合这些原则的。

5.1.4 分段数目影响分析

计算过程中我们发现分段数对 $\sum SP$ 存在影响。



图中，横轴为分段数(x 为 1 代表只允许直达)，纵轴为 $\sum SP$ 。可以看出分段数太大时， $\sum SP$ 将会下降；分段数为 2、3 时 $\sum SP$ 取值较大。

5.1.5 实例分析

以 D3022 次上海虹桥—武汉列车为例^[3]，获得 P、K 矩阵，算出 C 矩阵，通过分段售票模型下算出 SP 矩阵。

P	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	48	0														48
2	83	14	0													97
3	80	7	16	0												103
4	51	6	9	19	0											85
5	22	3	5	6	6	0										42
6	44	4	3	10	9	5	0									75
7	26	4	14	23	18	8	12	0								105
8	3	1	1	1	1	1	0	6	0							14
9	2	1	0	1	1	0	1	1	1	0						8
10	75	8	4	14	10	3	4	35	5	0	0					158
11	29	2	2	4	2	1	1	4	1	0	26	0				72
12	2	1	0	2	1	0	1	2	0	0	4	2	0			15
13	5	5	1	2	2	1	1	2	0	0	7	2	1	0		29
14	127	31	14	51	35	13	26	106	6	2	133	35	4	69	0	652
总计	597	87	69	133	85	32	46	156	13	2	170	39	5	69	0	1503

图 6 预测客流量 P 矩阵

K	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	44	0														44
2	98	15	0													113
3	80	7	15	0												102
4	49	5	11	25	0											90
5	18	2	3	8	7	0										38
6	34	5	5	11	11	9	0									75
7	29	5	15	26	21	6	14	0								116
8	2	0	1	1	0	0	0	5	0							9
9	2	0	0	0	0	0	1	0	0							3
10	66	7	5	11	9	2	6	46	4	0	0					156
11	27	4	3	3	2	1	0	5	1	0	27	0				73
12	2	1	0	1	1	0	0	0	0	0	5	1	0			11
13	5	4	2	2	1	1	1	3	0	0	3	1	1	0		24
14	103	30	21	66	33	13	24	111	5	0	122	32	9	61	0	630
总计	559	85	81	154	85	32	45	171	10	0	157	34	10	61	0	1484

图 7 实际客流量 K 矩阵

C	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	47	0														47
2	82	14	0													96
3	79	7	16	0												102
4	50	6	9	19	0											84
5	22	3	5	6	6	0										41
6	43	4	3	10	9	5	0									74
7	26	4	14	23	18	8	12	0								104
8	3	1	1	1	1	1	0	6	0							14
9	2	1	0	1	1	0	1	1	1	0						8
10	74	8	4	14	10	3	4	35	5	0	0					156
11	29	2	2	4	2	1	1	4	1	0	26	0				71
12	2	1	0	2	1	0	1	2	0	0	4	2	0			15
13	5	5	1	2	2	1	1	2	0	0	7	2	1	0		29
14	125	31	14	50	35	13	26	105	6	2	131	35	4	68	0	644
总计	589	86	68	131	84	32	45	154	13	2	168	39	5	68	0	1484

图 8 票额 C 矩阵

SP	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	8	0														8
2	83	7	0													90
3	80	7	10	0												97
4	51	6	9	6	0											72
5	22	3	5	6	1	0										37
6	44	4	3	10	9	0	0									70
7	26	4	14	23	18	8	8	0								101
8	3	1	1	1	1	1	0	1	0							9
9	2	1	0	1	1	0	1	1	0	0						7
10	75	8	4	14	10	3	4	35	1	0	0					154
11	29	2	2	4	2	1	1	4	1	0	21	0				67
12	2	1	0	2	1	0	1	2	0	0	4	0	0			13
13	5	5	1	2	2	1	1	2	0	0	7	0	0	0		26
14	127	31	14	51	35	13	26	106	6	1	133	34	3	65	0	645
总计	557	80	63	120	80	27	42	151	8	1	165	34	3	65	0	1396

图 9 允许分段，不限分段数 SP 矩阵

SP	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	47	0														47
2	83	13	0													96
3	80	7	15	0												102
4	51	6	9	18	0											84
5	22	3	5	6	6	0										42
6	44	4	3	10	9	4	0									74
7	26	4	14	23	18	8	12	0								105
8	3	1	1	1	1	1	0	6	0							14
9	2	1	0	1	1	0	1	1	0	0						7
10	75	8	4	14	10	3	4	35	5	0	0					158
11	29	2	2	4	2	1	1	4	1	0	26	0				72
12	2	1	0	2	1	0	1	2	0	0	4	2	0			15
13	5	5	1	2	2	1	1	2	0	0	7	2	1	0		29
14	127	31	14	51	35	13	26	106	6	2	133	32	4	68	0	648
总计	596	86	68	132	85	31	46	156	12	2	170	36	5	68	0	1493

图 10 允许分段，分段数为 2 的 SP 矩阵

$$a = \begin{cases} -88 < 0, & \text{允许分段, 不限分段数} \\ 9 > 0, & \text{允许分段, 分段数为 2} \end{cases}$$

比较图 7、9、10，以及两种情况下的 a ，可以看出：在不限制分段数的情况下，分段售票使成功出行总人数降低；而在限制只能分两段的情况下，成功出行总人数提高。综合 5.1.4 和 5.1.5，可见分段数限制为 2 或 3 较为合理。

5.2 问题二

5.2.1 公平且提高运力的情况分析

通过模型求解，我们认为存在这种允许分段购票条件下，既能保证公平又能提高运力的情况。

1、直达先购票，余票再分段

仍然使用 5.1.5 的数据，调整分段售票模型，使得只能购买直达票，不可分段购票，得到 P、C、SP 矩阵，再对此时的余票允许分段，最终得到 SP 矩阵。

SP	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	总计
0	0															0
1	47	0														47
2	82	14	0													96
3	79	7	16	0												102
4	50	6	9	19	0											84
5	22	3	5	6	6	0										42
6	43	4	3	10	9	5	0									74
7	26	4	14	23	18	8	12	0								105
8	3	1	1	1	1	1	0	6	0							14
9	2	1	0	1	1	0	1	1	1	0						8
10	74	8	4	14	10	3	4	35	5	0	0					157
11	29	2	2	4	2	1	1	4	1	0	26	0				72
12	2	1	0	2	1	0	1	2	0	0	4	2	0			15
13	5	5	1	2	2	1	1	2	0	0	7	2	1	0		29
14	125	31	14	50	35	13	26	105	6	2	131	35	4	68	0	645
总计	589	87	69	132	85	32	46	155	13	2	168	39	5	68	0	1490

图 11 直达先购票，余票再分段的 SP 矩阵

$$\text{直达先购票, 余票再分段} \begin{cases} a = 6 > 0 \\ b = \frac{6}{N} > 0 \end{cases}$$

可见，在直达能满足的所有客流购票完成后，放开分段售票，既保证了公平，又能提高运力。实际事务中，我们无法得知购买直达票的乘客是否已经全部购完，但是可用时间大致刻画，规定在开车前几天才启动分段售票，此时大部分直达购票乘客应已经完成购买。

2、限制分段数

继续使用 5.1.5 的数据。图 9 为允许分段，不限制分段数的 SP 矩阵，图 10 为允许分段，限制分段数为 2 的 SP 矩阵。

$$\text{允许分段，分段数为 2} \begin{cases} a = 9 > 0 \\ b = \frac{9}{N} > 0 \end{cases}$$

由 5.1.4 的分析，可知分段数对成功出行总人数有影响，分段太细会减少总出行人数，分段数为 2 或 3 时较为合理。本例限制分段数为 2，得到 a,b 均大于 0 的结果，保证公平性，同时提高运力。

5.2.2 客运运力资源调配系统复杂度和效率分析

1、算法复杂度、效率分析

本文的建模方法主要是对 P、C 矩阵的运算继而计算 SP 矩阵。下针对 5.1.5 的实例，分析算法复杂度。（程序见附录）

直达先购票，余票再分段情况，外层循环遍历 15 列，内层循环遍历 15 行，算法复杂度为 $O(N^2)$ 。但由于一辆列车上的站点数量很有限，多项式(N^2)级别的时间复杂度对问题的求解不会造成很大的影响

限制分段数为 2 的情况，外层同样循环遍历 15 列，中层同样循环遍历 15 列，内层循环为根据相应的组合情况处理不同的分段方案。Basic operation 为调用 Way()方法，所以程序的复杂度主要取决该方法的实现复杂度，Way()主要是使用相应的分段逻辑对 P、C、SP 进行相应的更改，实现时间复杂度为 $O(1)$ ，空间复杂度为 $O(N^2)$ 。内层循环的分段方式见 5.1.3，时间复杂度为 $O(N^3)$ 。同样的由于列车的站数很小，且最内层的循环里组合数的 m 和 n 都较小，因此多项式(N^3)级别的时间复杂度对问题的求解不会造成很大的影响

2、实务复杂度、效率分析

搜索系统：不允许分段售票情况下，只需搜索起始站间直达票额即可。而允许分段售票情况下，售票系统需要根据旅客的出发站、到达站搜索所有中间区间的余票，生成出行方案。限制分段数为 2 的条件下，搜索系统有两个步骤：搜索旅客出发站至中间站的票额，搜索中间站至到达站的票额。

售票存根：售票存根是对售票业务进行结账、清算、审核、营销分析的依据。分段售票情况下，存根应比直达票包含更多信息，包括出发站、中间站、到达站的各种信息。且系统存储记录也要相应包含更多信息^[2]。

5.3 公开信

致铁路总局客运部门的一封信

铁路总局客运部门：

您好！

近期，关于分段售票的议论很多，同车禁止分段购票的原因，并不具有说服力，而是否应该放开分段售票，更是说法不一。

我和队友在经过五天的数学建模研究后，想要告诉您我们的分析发现：

1.如果起售时即允许分段售票，会对客流总量造成负面影响。

2.但是，仍然存在允许分段售票，既能保证旅客购票公平性，又能提高客运运力的情况。即在开动前几天，大部分直达乘客需求已得到满足的情况下，放开分段售票可提高客流总量。同时，应注意限制分段数目，分段数在 2 或 3 比较合

理，分段过多会一定程度减少客流总量。

3.虽然分段售票会为售票系统以及实务管理增加一定复杂度，但是通过对算法复杂度的合理分析，我们认为这仍然是在可接受范围内的。

因此，我们的看法、建议是：可在售票末期开放分段售票，限制分段数为 2 或 3 段。

感谢拨冗阅读，祝铁道事业越来越好！

三名在校大学生

2019.5.1

六、模型推广

根据我们所建立的模型，可以看出对于不同的线路以及不同的列车具有普适性。不同点在于对于不同的线路，列车的站点数量可多可少，客流量所导致的票额分配也有一定程度上的波动，例如京广线的 G71 次列车分为 19 个站点，其运行路线长、客流量大的特点一定程度上引起 P、C、K 三个矩阵数据存在波动，这种情况下由于列车运行初期票额分配与出行人数相对正相关，仍然可以通过本文所述的求解模型对分段方式进行一定的规划。此外，本文模型的建立是为了解决铁路资源分配和客运运力优化的问题，通过分段方式这一宏观调控手段衡量分配关系从而在最大程度上得到铁路运行效率和客运资源调配的最优方案，决策者可以将各类影响因子如经济效益等放入规划模型中，进一步得到兼顾全局的最优解。该模型对于不同线路的铁路网均可逐段分析，对铁路管理部门可以作为分析解决问题的一种参考。

七、模型评价

7.1 模型优点

- 1.定义出衡量公平性的指标，使该抽象名词数值化。
2. 从复杂无序的情况中排列出可数个出行方案，呈现简洁、规律的结果。
- 3.最大程度保证长途旅客需求，稳定始发站的长途客流。

7.2 模型缺点

- 1.未考虑城市规模产生的影响。
- 2.未考虑旅客购票时间顺序产生的影响。
- 3.计算繁琐。

7.3 模型优化

针对缺点 1，设想通过给各站点设置权重的方法来改善。针对缺点 2，设想在售票系统网络环境下，通过及时更新数据以及计算机方法解决。针对缺点 3，设想通过完善、简化模型，使数据能通过计算机简单快捷算出。

八、参考文献

[1] 荣朝和. 如何科学统计高速铁路的载客水平与运营效率 [J]. 综合运

输,2010(09):66-68.

[2]史峰,周俊,陈东方,胡心磊,李竹君,李为康.同一列车不同席位分段拼接售票策略[J].铁道科学与工程学报,10(05):71-77,2013.

[3]王洪业,吕晓艳,周亮瑾,游雪松.基于客流预测的铁路旅客列车票额智能分配方法[J].中国铁道科学,34(03):128-132,2013.

[4]湛小丽.南京南站综合客运枢纽客流预测分析[J].智库时代,2018(30):157-158.

附录

```
clear
clc

cityName = ["上海虹桥" "昆山南" "苏州园区" "无锡" "常州" "丹阳" "镇江" "南京南" "
全椒" "肥东" "合肥" "六安" "金寨" "麻城北" "武昌"];

Row = 15;   Col = 15;

%% 意愿出行人数矩阵 - P
P = [
0      0      0      0      0      0      0      0      0
      0      0      0      0      0      0      0;
48      0      0      0      0      0      0      0      0
      0      0      0      0      0      0;
83     14      0      0      0      0      0      0      0
      0      0      0      0      0      0;
80      7     16      0      0      0      0      0      0
      0      0      0      0      0      0;
51      6      9     19      0      0      0      0      0
      0      0      0      0      0      0;
22      3      5      6      6      0      0      0      0
      0      0      0      0      0      0;
44      4      3     10      9      5      0      0      0
      0      0      0      0      0      0;
26      4     14     23     18      8     12      0      0
      0      0      0      0      0      0;
3       1      1      1      1      1      0      6      0
      0      0      0      0      0      0;
2       1      0      1      1      0      1      1      1
      0      0      0      0      0      0;
75      8      4     14     10      3      4     35      5
      0      0      0      0      0      0;
29      2      2      4      2      1      1      4      1
      0     26      0      0      0      0;]
```

```

2      1      0      2      1      0      1      2      0
      0      4      2      0      0      0;
5      5      1      2      2      1      1      2      0
      0      7      2      1      0      0;
127    31     14     51     35     13     26     106     6
      2     133     35     4      69     0;];

```

%% 票额矩阵 - C

```

C = [
0      0      0      0      0      0      0      0      0
      0      0      0      0      0      0      0;
47     0      0      0      0      0      0      0      0
      0      0      0      0      0      0;
82     14     0      0      0      0      0      0      0
      0      0      0      0      0      0;
79     7      16     0      0      0      0      0      0
      0      0      0      0      0      0;
50     6      9      19     0      0      0      0      0
      0      0      0      0      0      0;
22     3      5      6      6      0      0      0      0
      0      0      0      0      0      0;
43     4      3     10     9      5      0      0      0
      0      0      0      0      0      0;
26     4      14     23     18     8      12     0      0
      0      0      0      0      0      0;
3      1      1      1      1      1      0      6      0
      0      0      0      0      0      0;
2      1      0      1      1      0      1      1      1
      0      0      0      0      0      0;
74     8      4     14     10     3      4     35     5
      0      0      0      0      0      0;
29     2      2      4      2      1      1      4      1
      0     26     0      0      0      0;
2      1      0      2      1      0      1      2      0
      0      4      2      0      0      0;
5      5      1      2      2      1      1      2      0
      0      7      2      1      0      0;
125    31     14     50     35     13     26     105     6
      2     131     35     4      68     0;];

```

%% 成功出行人数矩阵 - SP

```
SP = zeros(Row, Col);
```

```
saveP=P;
```

```
saveC=C;
```



```

%% 模拟所有直达的人
SPrintf("模拟所有直达\n")
for xj = 1:1:Row
    for xi = Row:-1:xj+1
        if xi > xj
            t = min(P(xi,xj), C(xi,xj));
            SP(xi,xj) = SP(xi,xj) + t;
            P(xi,xj) = P(xi,xj) - t;
            C(xi,xj) = C(xi,xj) - t;
        end
    end
end

P=saveP;C=saveC;
xlswrite('result.xlsx',SP,1)
SP = zeros(Row, Col);

%% 最多只能分两段
SPrintf("最多只能分两段\n")
for xj = 1:1:Row
    for xi = Row:-1:xj+1
        if xi > xj

            [flag,C,P,SP] = Way(xj,xi,C,P,SP,xj,xi);

            if flag == 0
                for xk = xj+1:1:xi-1
                    if Way(xk,xi,C,P,SP,xj,xi) & Way(xj,xk,C,P,SP,xj,xi)
                        [flag,C,P,SP] = Way(xk,xi,C,P,SP,xj,xi);
                        [flag,C,P,SP] = Way(xj,xk,C,P,SP,xj,xi);
                    end
                end
            end
        end
    end
end

P=saveP;C=saveC;
xlswrite('result.xlsx',SP,2)
SP = zeros(Row, Col);

```

任意两座城市之间前往的情况

```
function [flag,newC,newP,newSP] = Way(source, dest, C, P, SP, solveS, solveD)
% 模拟从 source 城市到 dest 城市三个矩阵的变化情况
% source 为中间某段的出发城市，dest 为中间某段的目的地城市
% solveS 为出发城市，solveD 为目的城市
t = min([C(dest,source),P(solveD,solveS)]);

if t == 0
    flag = -1;
    newC = C;
    newP = P;
    newSP = SP;
    return;
end

SP(solveD,solveS) = SP(solveD,solveS) + t;
P(solveD,solveS) = P(solveD,solveS) - t;
C(dest,source) = C(dest,source) - t;

if P(solveD,solveS)>0
    flag = 0;
else
    flag = 1;
end

newC = C;
newP = P;
newSP = SP;
end
```