

基于博弈论和排队论的机场出租车调配模型

摘要

伴随着各大城市机场交通体系的不断建成与完善，如何对机场的陆地和公共交通进行调度成为了机场交通优化面临的主要问题。本文通过将博弈论和排队论的有机结合，对机场城市交通的参与者：出租车司机与乘客之间的博弈过程进行了模拟，探究各方选择与不同因素之间的联系。同时对机场道路的车辆运行情况进行模拟，利用上文探索出的车辆规律，为机场管理人员优化管理过程、提高载客效率提出了建议。

问题一、二 本题中我们通过博弈论视角，在双方信息透明的假设背景下，将司机视作单个的决策主体；而将落地的乘客抽象为同时决策的主体，通过层次分析法将影响其决策的主、客观因素量化表示。同时，我们将机场的具体道路和车辆通行情况用排队论模型进行模拟，以构建司机决策和乘客决策间的关系。在司机需要最大化收益的情况下，用某时刻的乘客数目表示出此时在机场等候的车辆数目阈值。对于满足理性人假设的司机来说，会自行比较车辆数目关系以决定是否等候。同时我们的模型中包含时间序列模型，此模型的存在允许我们输入某时刻的乘客到达量，并允许我们通过每个时间点的到达量自动推测出下一时刻的司机决策。此后我们在问题二中通过搜集上海虹桥机场近三天的乘客到达数，对模型进行了相应的验证。

问题三 本题中我们通过对并行车道的两种主要应用模式的构建，探究了双线式车道模型和交织车道模型在不同条件下的运输乘客能力。在双线式模型中，我们分别讨论单点式和矩阵式的情况，引入元胞自动机和车辆密度模型，分别探究其对于不同情况的适用性；而交织模型中，我们引入了交织车辆运行速度对其转换车道过程进行模拟。最终我们认为，双线矩阵式模型除了能保证较高的运输效率之外，安全性能也较好，故推荐使用该模型。

问题四 本题中我们通过尝试对短途车辆在蓄车池中的等待进行某种类类似于“插队”形式的补偿，允许一定范围内的插队比例，对行驶不同长度的车辆的时均收益的变化进行平均。最终我们的目的抽象为最小化其方差函数。由于其形式过于复杂但范围有限，我们用网格搜索发现，在调整不同参数的情况下，使其方差最小的补偿边界距离及“插队率”均较稳定地保持在一定范围之内。而为了避免可能欺诈的行为对模型造成的外部性影响，我们对相应的数值进行了修正，令其能更好地适应于现实情况的应用。

关键字： 博弈论 排队论 单目标优化 有限理性 时间序列

一、问题重述

伴随着交通系统的逐步完善和发展，飞机已经逐渐成为了长途旅行的首选交通方式。而在城市中奔驰不止的出租车则连接着城市与机场之间的距离。当出租车搭载着乘客来到了人流量稠密、道路复杂的大机场，出租车司机应该如何抉择来保证自己在获得最大的收益的同时为乘客们提供便捷的服务，是一个涉及各方之间关系、存在博弈的问题。

1. 研究影响出租车司机进行决策的机理，在司机的决策中将乘客的情况纳入考量，通过建立适当的模型帮助司机做出最优的策略。

2. 通过搜集某国内机场与其城市出租车运营情况的数据，综合分析司机的决策方案与其相关因素，同时对其合理性进行讨论。

3. 对于某机场的“乘车区”，若其中存在并行两条车道，应当如何对“上车点”及车辆和乘客进行安排，以在确保安全的前提下尽量提高机场的乘车效率？

4. 由于乘客目的地的远近存在差异，导致不同的司机在载客和回程的过程中可能获得不同的收益；但在不能选择乘客的情况下，管理部门希望通过给予一些短途返回的出租车以适当“优先权”以平衡这种收益差距。尝试提出合理的优化方案。

二、问题分析

本题是以博弈论为背景的策略选择和交通模型优化的综合问题。

2.1 问题一

由于在本题中司机进行的决策时对于“蓄车池”中的车辆数目和机场的乘客数都是明确的，而乘客对于交通工具的选择也是综合了自身情况的最优化判断。在双方的选择彼此关联的情况下，其中存在着明确的博弈关系。为了简化模型，我们首先对于司机空载回到市区的潜在收益用预设参数 α 进行了量化，也通过层次分析法对旅客效用进行了估计。在司机满足阈值条件选择通过等待载客的情况下，使用排队论模型对车辆及旅客数目关系进行分析，寻找旅客不同选择之间的效用均衡点以帮助司机决策。

2.2 问题二

本题中通过研究上海虹桥机场的相关数据，通过一天(白天)内不同时间的乘客数据对于出租车司机决策进行了预测；同时通过地铁、出租车的相关数据对上一问中模型的合理性进行了验证，对相关参数的依赖性进行了验证和估计。

2.3 问题三

本题中通过提过两种可能的安排模型：双线式车道模型与交织车道模型，对车道中是否允许车辆交织、如何安排两车道中的车辆做了规定。并分别通过元胞自动机的 NS 规则与相关的车辆运行规则及交织模型，对现实安排的车辆情况进行了模拟。同时对多停车点问题中 NS 规则的不适用性进行了证明，最终给出了一定条件下的最优解条件。

2.4 问题四

本题中，我们希望通过提出对运输短途旅客司机的补偿方案，使其获得的收益大致相等。由于总时间长度不同无法简单对总收益进行比较，我们选择对其平均时间内获得的收益进行对比。将方差作为最终标准的量度。最终得到在不同的长短途分界下，“允许插队”比例随之变化的关系。同时，通过调整在蓄车池中进入车道的方式，将短途车与长途车进行一定的区分放行以对其收益进行平均化，达到均一平均收益的目标。

三、模型假设

1. 出租车除去在“蓄车池”的等待时间之外，其他时间内运动的均速为 v km/h。
2. 由于地铁等其他公共交通方式夜间的运行情况及出租车定价标准会发生变化，故本题的模型中只考虑日间（09：00-21：00）的交通运行情况。
3. 出租车计价规则遵从《上海市出租车收费标准》；由于机场与市区间距离较大，故出租车单次行驶路程应大于三公里。
4. 乘客乘坐出租车以外的公共交通方式不需要额外的等待时间。
5. 出租车在蓄车池中等待时没有直接损失（包括油费、磨损等）。
6. 乘客乘坐地铁时，在其到达目的地前，地铁上的人数由于人员的上下车能保持相对稳定。
7. 国内到达航班视作中型飞机，根据波音 737 与空中客车 320 相关参数估计载客量为 150 人；国际/港澳台到达航班均视作大型飞机，根据波音 777 与空中客车 340 相关参数估计载客量为 350 人；所有出租车均视作上海大众朗逸车型，车长 $l_c = 4670\text{mm}$ 。
8. 每组私人游客的人数不超过四人（即只考虑私人乘客）。
9. 司机与乘客的决策均符合理性人假设。

四、名词解释和变量说明

变量符号	变量含义	变量单位
------	------	------

r	司机收益	CNY
$U_{s,t}$	乘客效用 (地铁, 出租车)	1
u	出租车起步价	CNY
z_1, z_2	出租车每公里单价, 远程单价	CNY/km
α	出租车在市区内载有乘客的概率	1
β	每公里耗油率	L/km
p_o	单位油价	CNY/L
d	出租车回到市区的行驶路程	km
t_w	等待时间	h
t_p	单个旅客等待时间	h
l	舒适程度	1
γ	疲倦系数 (时间)	h^{-1}
ω	疲倦系数 (人)	-1
w	天气	1
n	同行人数	1
s_1, s_2, s_3	起步里程标准, 远处里程标准, 城区范围阈值	km
v	出租车的平均运动速度	km/h
v_0, v_1	地铁均速, 行人速度	km/h
c	乘客的总支出	CNY
c_0	地铁票价	CNY
n_0	落地旅客人数	1
n_1	地铁乘坐人数	1
t_1	乘客到达间隔	h
$a_j(j = 0 \dots n)$	系统滞留车数	1
$f_j(j = 0 \dots n)$	系统输入车数	1
$b_j(j = 0 \dots n)$	系统滞留人数	1
$g_j(j = 0 \dots n)$	系统输入人数	1
t_s	单服务台系统中服务时间	h
r_h	单位时间内的平均收益	CNY/h
s_4	短途司机的补偿边界	km
P	出租车司机接到短途行程的概率	1
ζ	对短途行程出租车提前安置的比例	1
L	进入上车点的车道长度	km
k	离开车道的车流量密度	h^{-1}

五、问题一模型的建立与求解

5.1 司机的最大收益选择

在问题一中，我们帮助司机做出决策时，影响其决策的核心要素是其收益。根据上文中已经定义参数及出租车计价规则，若司机选择载客回到市区，其收益为：

$$\begin{cases} r = u + (d - s_1) * z_1 - d * \beta p_o, s_1 \leq d \leq s_2 \\ r = u + (s_2 - s_1) * z_1 + (d - s_2) * z_2 - d * \beta p_o, s_2 \leq d \end{cases} \quad (1)$$

而司机若选择空载，则其即便会损失空载回程的费用，但是其在市区内有更大的概率载到乘客。出于简化模型的考虑，我们假设司机回到市区内后“载有”乘客的概率为 $\alpha(0 < \alpha < 1)$ 。则此时我们可以将司机空载回程后与完成等待的同等时间内的收益表示为：

$$\begin{cases} r = \alpha[u + (v * t_w - s_1)z_1] - (d + vt_w)p_o, s_1 \leq d \leq s_2 \\ r = \alpha[u + (v * t_w - s_2)z_2 + (s_2 - s_1)z_1] - (d + vt_w)p_o, s_2 \leq d \end{cases} \quad (2)$$

需要注意的是，由于司机对于乘客的乘坐距离只有在乘客上车之后才知晓，且不考虑司机拒载的情况，因此我们在程序中计算司机收益时，用旅客乘坐距离的数学期望带入 d 的运算。

5.2 旅客的交通方式选择

由于变量 d 是由乘客决定、且影响着乘客交通工具选择的变量，在决定 d 时我们同时需要对乘客选择的交通方式进行模拟。又因为在乘客选择交通方式时，影响方式较为多元。我们在评判时选取经济学中的“效用”(U)概念作为乘客的决断依据。而为了使我们对效用的评价更加全面、客观，我们在选取相应的评判指标时选取出了五个方面的变量进行评判。由于效用概念的主观性，我们采取了层次分析法对其进行分析，以对各影响因素的影响权重进行比较，同时量化乘客在选择交通工具中的一些主观性部分。

下面对其中的一些影响因素进行详细定义：

5.2.1 总时间

乘客在选择不同交通方式时，快捷是其首要考虑的要素。在本题中我们将乘客的总时间花费定义为旅客等待时间与交通时间的和，即：

$$t = t_p + \frac{d}{v} \quad (3)$$

而由模型假设，

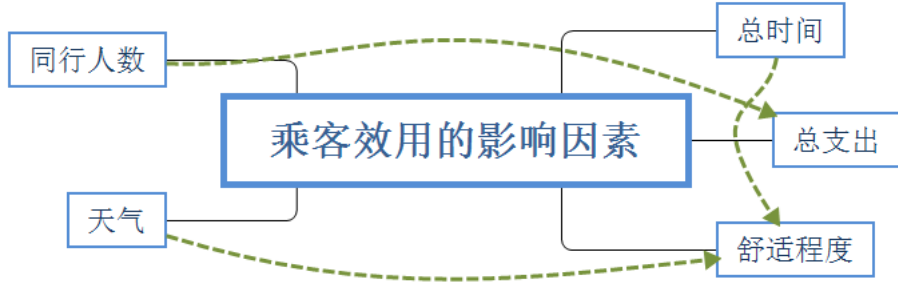


图 1 乘客效用的影响因素

$$t_p = [b_0 + (n_0 - n_1)] * t_s \quad (4)$$

由于已经假设过地铁需要花费的等待时间为 0，而地铁在地下运行时不能采取“直达”的方式，但不同的运动距离的位置偏差量影响程度不一，我们需要在地铁速度的计算中予以修正。

5.2.2 总支出

由于在乘客决策中已经考虑到时间支出，因此在对乘客的总支出进行评判时只需计算乘客的会计支出 (Account Cost)，即无需考虑其潜在的损失：

$$\begin{cases} c = u + (d - s_1) * z_1, s_1 \leq d \leq s_2 \\ c = u + (s_2 - s_1) * z_1 + (d - s_2) * z_2, s_2 \leq d \leq s_4 \end{cases} \quad (5)$$

5.2.3 同行人数

不同的同行人数对于乘客的乘车选择的影响体现在：一方面，更多的同行人数激励其享有出租车中更专有的空间；另一方面，更多的同行人数对出租车的乘车费用进行分摊，以降低总支出。故在计算总支出时需要对其进行进一步分摊。而在同行人数恰好等于出租车满载人数 (即四人时)，乘客具有最高的意愿乘坐出租车。

5.2.4 天气

天气情况对于乘客交通方式的选择有直接影响。在雨天乘客的出行具有更高的负效用。综合题目中的季节性考量，我们直接用某城市单个月份的降雨率表示其天气参数 w ，通过对其在不同交通方式中给予不同的权重补偿对其进行评判。

$$w = \text{Rainfall Rate}$$

5.2.5 舒适程度

舒适程度是一个相对主观的概念，在本模型中我们通过将其量化为乘坐同一交通工具的人数和乘坐时间的函数，即：

$$l(\frac{d}{v}, n_1) = (\exp(-\gamma * \frac{d}{v}))(\exp(-\omega * n_1)), 0 < \gamma < 1, 0 < \omega < 1 \quad (6)$$

在此，由于舒适程度概念的主观性和难以比较性，我们将其用上述函数控制在 0-1 范围内，以方便对其进行后续比较。该函数对于乘坐时间和具体人数情况完全呈负相关，与现实情况能够较好地符合。

通过对乘客的现实偏好的评判，我们通过层次分析法得出下列的判别矩阵：

$$\begin{bmatrix} 1 & \frac{4}{5} & 7 & \frac{10}{11} & 2 \\ \frac{5}{4} & 1 & 9 & 2 & 5 \\ \frac{1}{7} & \frac{1}{9} & 1 & \frac{1}{5} & \frac{1}{3} \\ \frac{11}{10} & \frac{1}{2} & 5 & 1 & 2 \\ \frac{1}{2} & \frac{1}{5} & 3 & \frac{1}{2} & 1 \end{bmatrix} \quad (7)$$

计算出的一致性

$$CR = \frac{CI}{RI(n)} = 0.0124 < 0.1 \quad (8)$$

满足一致化检验。再对其归一化处理后得出其相应的权重为：

$$[w_1, w_2, w_3, w_4, w_5] = [0.2449, 0.3967, 0.0387, 0.2148, 0.1049]$$

观察权重结果发现，同行人数的比较权重比较低，低于 4%，而其运算设计参数较多，较为复杂，因此我们将其忽略，重新对于另外四者的权重进行计算，得更新后的权重为：

$$[w_1, w_2, w_4, w_5] = [0.2548, 0.4127, 0.2234, 0.1091] \quad (9)$$

再通过两种交通方式之间的相互比较，得到当乘客选择乘坐出租车的效用高于乘坐地铁的效用时，当且仅当 (由于时间与花费对于交通方式选择的影响是负向性，故采用除号)：

$$C = w_1 / \frac{t_p + \frac{d}{v}}{\frac{d}{v_0}} + w_2 / \frac{\frac{c}{n}}{c_0} + w_4 * \frac{w}{1 - w} + w_5 * \frac{l(\frac{d}{v}, 0)}{l(\frac{d}{v}, n_1)} \geq 1 \quad (10)$$

时，乘客具有更高的意愿乘坐出租车而非乘坐地铁。

5.3 出租车和地铁的人数模型——通过排队论模型和时间序列的模拟

在上述模型中， α, v, s 等均为已预设的常量， d 为由乘客决定的变量 (在程序实现中我们通过其正态分布用蒙特卡罗方法进行实际模拟以帮助计算)，而 t_w 则由机场中蓄车

池内车辆和机场乘客情况决定。在本问中，我们选取单个接客点为例，通过单服务台的排队论模型对其人数规律进行探究，并通过其与博弈论模型的综合判定得出司机的决策方式。

出于模型简化的需要，我们在模拟飞机和乘客到达时，将某个时间间隔 t 内到达的乘客数合并统计。即，在设定的时间间隔节点，半个小时内到达的旅客同时博弈。由于最终我们不代表乘客进行预测，故不需要分析每个乘客的交通方式选择方式，而可根据现场的人数、天气、目的地的距离长短，在宏观上分析出选择不同方式的乘客比例。在这种情况下，司机可以根据观察到的乘客数量和前方等待的车辆数目，衡量进入蓄车池的收益。当边际收益小于回到市区的潜在收益时，司机则不会进入系统。此时蓄车池达到车辆阈值。

而根据这种方式构建的模型，在每一个统计的间隔点处，对于决策主体司机和乘客来说，其决策方式只与上一个时间点的结束状态有关。故在此我们引入时间序列对研究范围内的时间状态进行分析。

5.3.1 某时间点的排队模型

在某时刻，系统中原有人数 b_j (组数，即同行的一组人)，原有车数 a_j ，单服务台的服务时间为 t_s 。该时间点到达人数为 g_j 。在该单服务台模型中，已知每个已经进入系统的主体具有足够的理性能力和充足的信息，故已在系统中的主体没有偏离的倾向。而每次服务耗时 t_s 后，系统中车辆数量相应减少，人数则相应减少一组。故司机等待时间：

$$t_w = a_j * t_s$$

, 乘客等待时间：

$$t_p = (b_j + g_j) * t_s$$

(司机是此时的单独决策者，故不需要考虑此时的同时决策者)

而对于本题研究单辆出租车博弈中的最优策略时，博弈的另一方乘客们具有两个显著特征：

- 1) 同时博弈
- 2) 信息透明

此时在无需考虑单个个体的具体选择的情况下，理性 (rational) 的乘客们在宏观规律上会自然趋向于均衡。即，选择地铁与出租车的收益相同，且双方都不存在改变自己选择的动力。故有：

$$U_s = U_t. \quad (11)$$

相应的，我们可以得到以下的方程：

$$C = 1 \quad (12)$$

根据上述的硬约束条件，我们可以解出在一定预设参数下选择不同交通方式的人数。根据人数与直流车辆的关系可以解出出租车的等待时间，由此解出相应情况下出租车司机的选择。

5.3.2 带入时间序列的综合决策模型

在本题的模型中，大部分的参数都是有定值的常参数。理论上给定每个起始点的旅客量与每个间隔后到达的客流量，以及初始状态的汽车量即可计算出一个时间范围内的出租车选择，从而迭代出一整天的出租车司机及旅客决策。而我们观察到其中唯一没有给定的量，即到达客流量，根据一天内具体时间的不同每天具有一定的周期性。根据搜集到的上海两大机场的相关数据，我们发现每半个小时时间内的乘客到达数量比较零散，不具有较好的线性特征，故而移动平均法与指数平滑等方法均不适用于本模型。但，由于只要根据初始时刻人数数据，模型中其他参量都为定制，因此我们可以在每一计数时间间隔点输入离散的客流变量，并做出下一时间点的相应决断。

综合以上所有条件，我们得出司机的选择在机场载客的条件为：

$$u + (d - s_1)z_1 - dp_o \leq \alpha[u + (vt_w - s_1)z_1] - (d + vt_w)p_o, (s_1 \leq d \leq s_2) \quad (13)$$

或：

$$u + (s_2 - s_1)z_1 + (d - s_2)z_2 - dp_o \leq \alpha[u + (vt_w - s_2)z_2 + (s_2 - s_1)z_1] - (d + vt_w)p_o, (s_2 \leq d) \quad (14)$$

s.t.

$$w_1 / \frac{t_p + \frac{d}{v}}{\frac{d}{v_0}} + w_2 / \frac{\frac{c}{n}}{c_0} + w_4 \frac{w}{1 - w} + w_5 \frac{l(\frac{d}{v}, 0)}{l(\frac{d}{v}, n_1)} = 1 \quad (15)$$

否则司机会选择空载回到市区。

(需要注意的是， a_j 在此代表单个服务台中累积的车数，如果具体的机场模型中服务台数量不为 1，则其需要除以适当的系数； t_p 中的 b_j, g_j 随时间序列更新)

六、问题二的求解与分析

6.1 以虹桥机场旅客数据为例进行的分析

根据现实情况和相关数据分析，我们对以下的参数进行了估计： $U = 14, z_1 = 2.5, z_2 = 3.6, \alpha = 0.7, \beta = 0.056, p_o = 6.66, \gamma = 0.05, \omega = \frac{1}{640}, v = 40, s_1 = 3, s_2 = 10, s_3 = 50, v_0 = 30, c_0 = 6, a_0 = 60, b_0 = 20, t_s = 0.0042$ 。平均每辆车载客 1.28 人。另外，由于平峰时段虹桥机场采取 8*1 的斜列式停车方式，故将其视为八个独立服务台系统。[1]

(相关数据还来自：上海市出租车收费标准，2019 年 9 月 10 日 92 号汽油价格调整，大众朗逸标准款车辆数据)

通过分析上海市地理环境和乘客在机场乘坐出租车的相关数据，我们将乘客的乘坐距离函数表示为 $d \sim N(24, 49)$ 。此时根据 3σ 原则，可以发现分布在其中的乘坐距离为 3-45 公里，符合上海实际市区与乘客需求的情况。

同时我们搜集了近三天上海虹桥机场的落地时间和出租车流量的平均数据，将每半个小时作为一段时间进行计算，用 24 个离散数据替代每一时刻的到达乘客数。用其算出了每一时间段内到达乘客的分布。在明确乘客分布的基础上，可以研究出在某一具体时刻系统中的出租车数。即在当前人数安排情况下，若进入更多的出租车，新的出租车获得的收益会低于空载状态。故在如下阈值下的出租车司机会进入机场排队，而超过其的司机则选择空载。

到达的人数 (从 9: 00-21: 00):

[1200, 1500, 1200, 2400, 1700, 1800, 1900, 2200, 2150, 1800, 1500, 2000,
1850, 1850, 2400, 1900, 2800, 1050, 1700, 1350, 2150, 1350, 2250, 1350]

每一时间点的车辆阈值 (总车数，对于单个服务台需要分别除以 8):

[1687, 1593, 1787, 1143, 1293, 1387, 1318, 1143, 1143, 1356, 1593, 1249,
1356, 1356, 1143, 1143, 1143, 1406, 1462, 1687, 1143, 1687, 1143, 1624]

当在相应时间点司机观察到车辆超过对应的数值时便不再进入候车池等候。

6.2 对相关因素合理性和依赖性的探讨

在研究了相关数据及结果后，我们发现最终的车辆数目相对较为符合现实情况。平峰时段单个候车口的等待车辆在 150-200 左右，等待时间在 30 分钟左右，较为合理。在计价规则、旅客到达情况相对稳定的情况下，我们对城市载客率和服务台服务时间这两个最为重要的参数的影响进行了验证。

通过对城市载客率 α 的检验，我们将其的取值在 0.5-0.7 之间的变化列出，发现在不同参数下，其随着时间变化趋势完全相同，而蓄车池车辆阈值随着 α 的增大单调递减。在市区载到乘客概率减小的情况下，司机倾向于来到机场排队等候，其数据表现符合常理，随之的变化趋势也足够大。(见图 2、3)

而对于服务时间 t_s ，我们遵从 [1] 中的时间模型，将具体服务时间拓展到平均值到两倍标准差范围 ($11.10+2*4.07 \approx 19s$) 内进行验证，发现等待出租车数随着服务时间的减小，趋势变化大致相同。而具体等待车数则与服务时间严格正相关，同样符合题目的假设。综合以上条件，我们认为我们的模型对实际情况有较好的模拟。

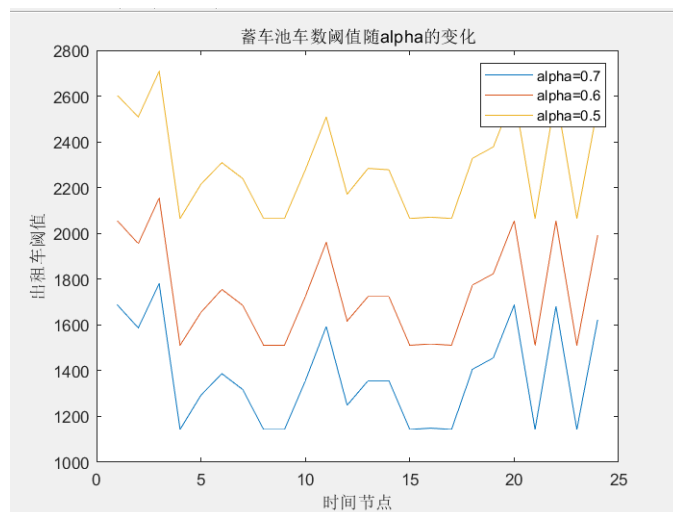


图 2 蓄车池车数阈值随 α 的变化

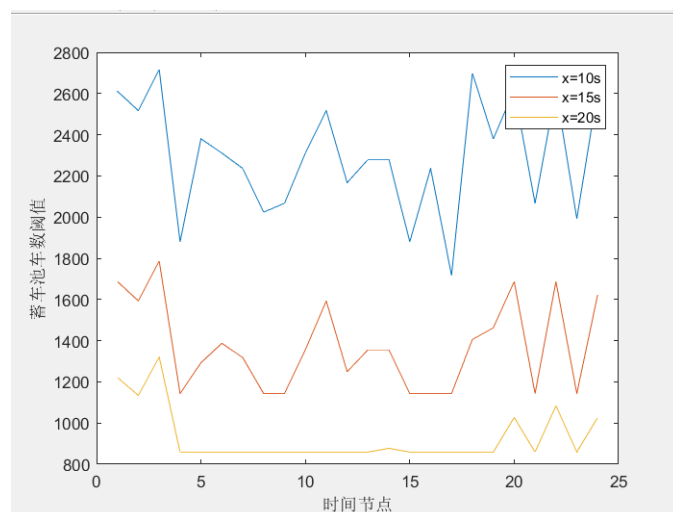


图 3 蓄车池车数阈值随 t_s 的变化

七、问题三模型的建立与求解

在本题的两条并行车道模型中，我们将道路模型大致的区分为两种形式：双线式和交织式。其核心区别为：双线式道路上的机动车无法变道，只能在既定道路上行驶；而在交织式道路上允许车流交汇。下面我们对其可行性分别进行讨论。

7.1 双线式车道模型

本题中我们需要优化的目标是单位时间内离开的车辆数目，即 k 。在双线式车道中，出于安全性考虑，我们用元胞自动机的 NS 规则来模拟车辆的运行规律。其基本原理是将实际道路抽象为长度为 L 的一维离散格点链，在任意时刻，每个格点有两种状态，为空或被车辆占据。该模型包括 4 步更新准则：

- 1) 加速: $v_n \rightarrow \min(v_{n+1}, v_{max})$;

- 2) 减速: $v_n \rightarrow \min(v_n, x_{n+1}, x_{n-1})$;
- 3) 随机慢化: 以随机概率 p 令 $v_n \rightarrow \max(v_{n-1}, 0)$;
- 4) 位置更新: $x_n \rightarrow x_n + v_n$

准则中, v_n 为第 n 辆车的运行速度, x_n 为该车的位置, v_{max} 为最大速度, l (即 l_s) 表示车辆长度, 通常取 1, p 为车辆随机慢化概率。准则 1) 体现出司机通常会以尽可能大的速度行驶车辆。准则 2) 反映车辆减速过程, 可保证所有车辆不会追尾前车。准则 3) 是道路自发产生堵车的至关重要因素, 不仅反映了驾驶员在减速过程中的过度反应行为, 也反映了驾驶过程中的随机减速行为。该准则借助随机慢化很好地体现了驾驶员的行为差异。[2]

我们根据上述建议, 将网格宽度视作 $l_s=7.5m$ 。

下面我们分别讨论, 单点式和矩阵式的设置情况。

7.1.1 单点式 (2*1)

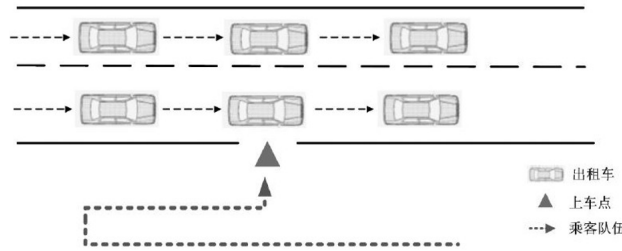


图 4 单点式模型

在只有单个上车点时, 系统每次更新时间 (步长) 为移动一个方格的最短时间。结合系统中相应的速度及为了与服务时间配合, 我们选择时间步长 $t_a = 2.5s$ 。在此情况下, 元胞自动机一定可以维持在下状态: 即在前一辆车一离开车位时, 下一辆车在下一个时间步长可以进入上车点并提供服务。而由于不存在累积车流, 旅客上车和去向道路另一侧 (过马路) 不需要额外时间。故此时, 每过 $t_s + t_a$ 就有两辆车离开系统。在这种情况下, 离开的车流量密度为 $\frac{2}{t_s + t_a} \approx 411$ 辆/小时。其是否为最优解尚不明晰, 下面我们分析矩阵式情况。

7.1.2 矩阵式 (2*X)

由于矩阵式模型中, 旅客到达位置是统一的, 故旅客前往不同的上车点有不同的时间需求; 又因为具有两条车道, 所以还需要在系统中预留乘客过马路的时间。而同时在本模型中, 乘客需要步行的需求导致车辆间的距离需要尽量较小, 故机场管理人员需要一次放入一定数量的车辆令其以慢速行驶, 同时停在多个邻近的上车点的位置。而旅客需要在到达上车处后分别前往上车点。

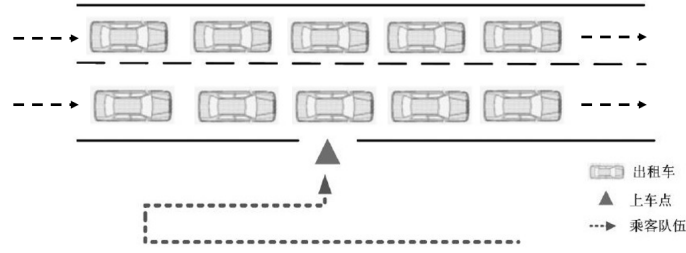


图 5 矩阵式模型 (2*X)

在矩阵式模型中，我们在模拟过程中发现，NS 规则的模拟遇到了几个问题：

1) 只要前一个位置的车不运动，下一辆车即不能运动；而当矩阵式模型运行时，我们要求不同的车辆在同一时刻停在相邻网格中。而这对于我们元胞自动机的 NS 规则是没办法实现的。

2) 元胞自动机中的网格长度较长，对于题目中给出的条件，每辆车之间间距达到 $(l_s - l_c) \geq 2.8m$ 。其中停下的车辆之间距离过长，同时需求的跑道长度过长，对于现实空间没办法有效利用，尤其在人流量稠密的机场。故模拟过程中存在局限性。

故我们在矩阵式模拟的过程中对模型进行了修正：

与此同时，由于脱离了 NS 规则，为了保证车辆的安全，系统中的车辆密度和车辆速度应当控制在一定的范围之内。经过查阅文献，我们知道由交通流理论可知行车速度与车流密度密切相关，根据 Greenshields 提出的速度—密度关系模型：

$$V = V_f(1 - \frac{\rho}{\rho_j}) \quad (16)$$

其中， V 为行车速度； V_f 为畅行速度，车流密度趋于零，车辆可以畅行无阻时的平均速度； ρ 为交通流密度； ρ_j 为阻塞密度，即车流密度大到车辆无法移动时的密度。[3]

根据以下数据，我们建立了相应的模型：

人行速度 $v_1 = 5km/h$, 马路宽度 $L_m = 9m$, 在上车点处紧密条件下两车间排列的距离为 $l_t = 0.15m$, 车道长度为 $50m$, V_f 取 $20m/s$, 阻塞密度 ρ_j 为 $5.5m^{-1}$ 。在每个运动时刻的车速记作 v_2 。

得出在上车点数为 X 时，其车辆通行效率 k 与 X 的关系为：

$$k = \frac{2X}{((t_s + t_g + \frac{l_c}{v_2}) + \frac{(X-1)l_c}{2v_1})} \quad (17)$$

综合两方程，取其整数解。结合经验判断解在 5 或者 6。算出其在 5、6 的值分别为 (5,926.597),(6,928.350)。在这两种情况下通行能力差距极小。故每次应当允许进入 10 或 12 辆车，得到的每小时最大通行车辆数在 928 辆左右。

7.1.3 对于双线式车道模型比较的修正

在双线模型中，对于不同的上车点情况，我们分别采取了 NS 规则和结合了道路流的速度、密度关系方程对最大运载效率进行了估计。在矩阵式情况下我们抛弃了元胞自动机。而若是将元胞自动机在单停车点的车辆间距进行修正，则得到的运载效率应当在 $\frac{411(2l_t+l_c)}{l_s} \approx 612$ 辆左右，而此种修正仍然无法修复问题 1)，故我们在上述现实条件下，在双线式车道模型中得到的最优解即为每次容许 10 或 12 辆车进入，平均每 4 秒左右发一辆车。具有相当好的运动属性，也能大致反应现实情况。故我们认为修正后的双线式车道模型具有一定的稳定性和现实性意义。

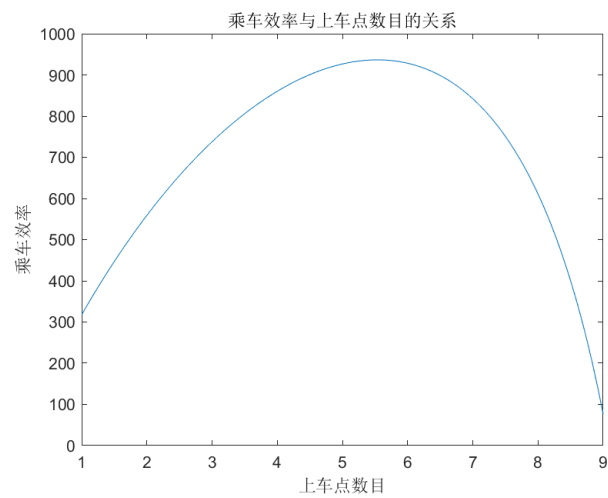


图 6 乘车效率和上车停车点数量的关系

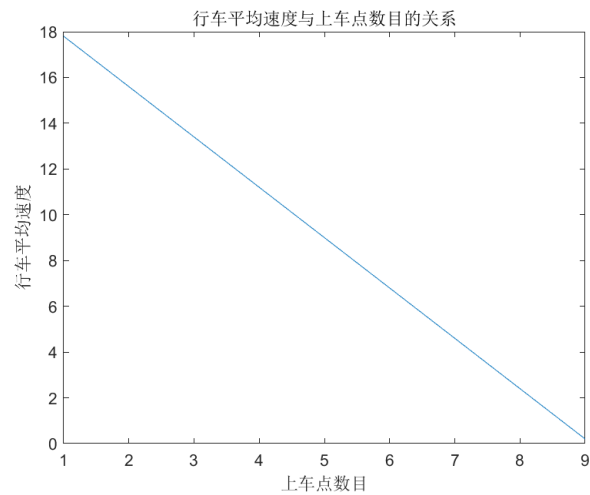


图 7 行车平均速度与上车点数量的关系

7.2 交织车道模型

7.2.1 模型详情

多点纵列式出租车排队服务系统属于面向乘客的带有多个服务台和一个公共队伍的排队系统。乘客按照一定的到达规律到达排队系统后,排在队伍前端的乘客可以根据当前上车点的出租车服务状态分散到纵向排列的多个“服务台”接受服务。出租车则由内侧车道驶入港湾式上车点进行载客服务,服务结束后驶离上车点,然后由后面的出租车补位。此类出租车排队系统与单点式出租车排队服务系统相比,增加了上车服务点,提高了系统的服务效率,但当多个服务点的出租车同时驶离各自的上车点时,车辆之间易产生干扰与冲突。此外,此类出租车排队系统对纵向空间长度要求较高。[4]

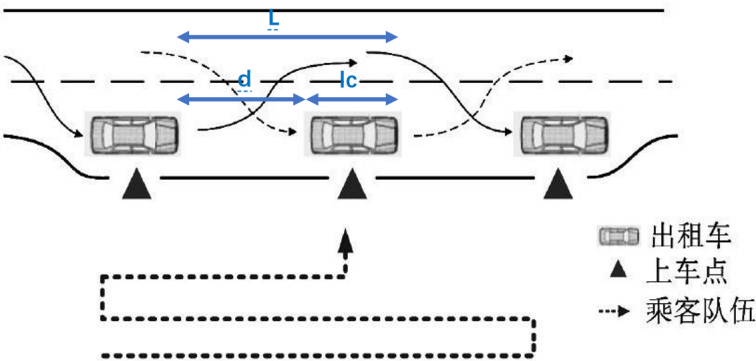


图 8 交织车道模型图示

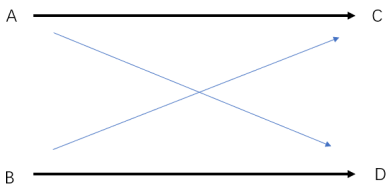


图 9 交织区示意

概念：两股或多股交通流在没有交通控制设施的情况下，沿着相同的大方向在相当长的公路路段中运行看，其中相交而过的交通流成为交织。一般分为 A、B、C 三种类型。在图 9 中，A->D,B->C 属于交织车流，A->C,B->D 属于非交织车流。发生交织运行区域为交织区。A 类交织区要求每个交织车辆进行一次车道交换。

此处模型属于 A 类交织区。A 类交织区中能被交织车辆使用的最大车道数是最受限制的。一般交织车辆将它们限制在邻接路拱线的两车道之中，故不论有用的车道数是多少，交织车辆一般最多用到 1.4 车道。[5] 故为确保出租车停靠的安全性和可行性，此处交织段运行状态必须处于约束运行，需满足下式： $N_w \geq N_w(max)$, $N_w(max) = 1.4$

即:

$$\frac{1.21 * N * (VR)^{0.571} * L^{0.234}}{S_w^{0.438}} \leq 1.4 \quad (18)$$

N_w —交织车辆为了取得均衡运行或非约束运行必须采用的车道数。

$N_w(max)$ —对于指定类型交织区, 交织车辆可采用的最大车道数。

式中, L 为交织区长度 $=l_c + d_g$ 米, 其中 l_c 为车长 (朗逸 4.7 米), d_g 为服务台之间间隔, 即两辆停泊的出租车之间的间隔。N 为车道数, VR 为交织流量比, 此处由于所有车辆都要进行交织, 所以取值为 1, S_w 为交织车辆约束运行状态下运行速度, 单位 km/h。

设有 n 个服务台。每辆车经过最后一个服务台到第一个服务台的时间可分为两次经历交织区的时间 $t_1 = 2 \frac{l_c + d_g}{S_w}$, 服务时间 $t_2 = t_s$, 外车道行驶时间 $t_3 = \frac{(X-2)l_c + (X-3)d_g}{v}$, v 为出租车外车道行驶速度, 乘客从排队出口走到出租车上的平均时间 $t_4 = \frac{(n * l_c + (X-1) * d_g)}{(4 * v_0)}$, v_0 为行人速度。

那么单位时间内通过此系统的车辆数为:

$$k = \frac{n}{t} = \frac{n}{t_1 + t_2 + t_3 + t_4} \quad (19)$$

此处交织车辆运动速度 S_w 小于外车道行驶速度, 即 $S_w \leq v = 20\text{km/h}$ 。生活经验可得 $d_g \geq 4\text{m}$, 出租车司机才有可能一次顺利停车。在上述不等式成立下, 求解 $\frac{n}{t}$ 的最大值。由于本函数中具有三个变量, 我们先固定 n , 寻找函数随 d 和 s_w 之间的变化关系。但又由于在两模型进行比较时总车道长度应与双线式模型中应当统一, 故应当满足:

$$l_c X + (X - 1)d \leq 50 \quad (20)$$

取 $l_c = 4.7\text{m}$, $v=20\text{km/h}$, $v_0 = 5\text{km/h}$, $t_s=15\text{s}$, 求解得出, $d_g=4\text{m}$, $n=6$, $k = (\frac{X}{t})_{max} = 665.1$ 辆/小时。 d_g 取值越小, 即出租车停靠越密集, 乘车效率更高。

7.2.2 对于交织车道模型合理性的检验

在上文的模型证明中, 我们先验地假设出租车必须一次进入停车位, 并因此给予两车之间一个较大的间隔。如果我们在本模型中为了提高车辆密度以增进其运输效率, 并允许其在车位上进行一定类似倒车的操作, 并对这种倒车操作的时间给予一定补偿, 则我们可以将模型中的参数调整为: $n=7$, $t_s=20\text{s}$, 车辆间距离 $d_g=2\text{m}$ 时, 得出结果为: $\frac{n}{t} = 693.7$ 辆/小时。

可以清楚地观察到, 在降低车量间距、提高危险性和事故几率的情况下, 最终的车辆输出效率的确有所升高。由此可以验证本题中我们建立的模型在留足转弯空隙确保安全的情况下, 降低车辆间距、提高密度可以在一定程度上提高通行效率。据此我们基本可以断定本交织车道模型的合理性。

7.3 机场车辆安排模式的最终形式

通过上面两个部分的分析，我们发现对于上面分类的两种情况中，不允许车辆变道的双线式车辆模型在具有更高安全性的前提下，在现实情况的运用中同样具有更高的载客效率。故我们建议机场应当选择相关模型进行运用。而除去这种安排模式，机场同样可以通过设置人力帮助乘客上车缩短上车时间 (t_s) 等方式对现实机场情况进行进一步优化，为乘客带来更优质的服务。

八、问题四模型的建立与求解

8.1 按比例分配长短途出租车进入车道的优化方法

在设计题目中需求的补偿机制时，为了对短途司机的效益有所补偿，我们必然需要允许一定的“插队”行为；而为了保证公平，同时不让优先级具有对长途司机的损失具有过大的影响，其补偿机制必然是有限制的。因此，我们选择采取按比例分配短途出租车进入车道的方法。简单的说，我们为短途车司机提供一条新的道路，使其重新回到机场时进入候车区所需要排队的时间与正常排队需要的总时间成一定的比例 ζ 。按照这种思想我们建立相应的模型。

对于本题中的目标，即平均化收益，我们首先需要对时间范围进行限制。此时由于涉及了下一次载客的顺序，故在考虑是我们必须使司机存在动力再次载客回到机场。故我们需要计算入司机回到机场的时间。同时，司机回到机场的收益应当大于其在市区载客的收益，故我们的目标优化为 (D 代表平均收益方差)：

$$\min D(r_h) = \frac{1}{s_3 - s_1} \int_{s_1}^{s_3} \left[r_h - \frac{\int_{s_1}^{s_3} r_h d(d)}{s_3 - s_1} \right]^2 d(d) \quad (21)$$

其中，在引入不同的行驶距离和需要引入的短途边界之后，我们按照一定比例，为短途的出租车提供优先离开蓄车池进入车道排队的优先级。据此我们对于其中的均值做如下的修正 (E 代表数学期望)：

$$\begin{aligned} E = & \frac{1}{s_3 - s_1} \left\{ P \left[\int_{s_1}^{s_2} \frac{u + (d - s_1)z_1 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} d(d) \right. \right. \\ & + \left. \int_{s_2}^{s_4} \frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} d(d) \right] \\ & + (1 - P) \left[\int_{s_4}^{s_3} \frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j (2 - \zeta) + 2\frac{d}{v}} d(d) \right] \left. \right\} \end{aligned} \quad (22)$$

同样的，若按照这种方式提供优先级，则我们可以将方差进行如下的修正：

$$\begin{aligned}
\min D(\zeta, s_4) = & \frac{1}{s_3 - s_1} \{ P [\int_{s_1}^{s_2} (\frac{u + (d - s_1)z_1 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} - E)^2 d(d) + \\
& \int_{s_2}^{s_4} (\frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} - E)^2 d(d)] \\
& + (1 - P) [\int_{s_4}^{s_3} (\frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j (2 - \zeta) + 2\frac{d}{v}} - E)^2 d(d)] \}
\end{aligned} \quad (23)$$

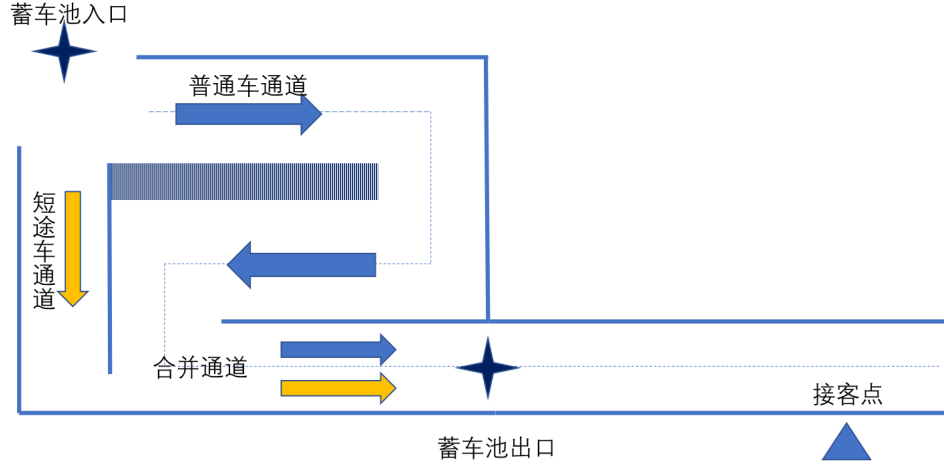


图 10 插队模型示例

(其中橘色线路：蓝色线路容纳车数比例为 ζ)

同时在实际操作中为了避免长途司机的损失过多，我们为其设置了一个转移阈值条件，令其最后平均时间内获得的收益不能小于短途的 90%(否则会导致司机只愿意搭载短途乘客)。数学表示为：

$$\begin{aligned}
(1 - P) [\int_{s_4}^{s_3} \frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j (2 - \zeta) + 2\frac{d}{v}} d(d)] \geq \\
0.9P [\int_{s_2}^{s_4} \frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} d(d)]
\end{aligned} \quad (24)$$

$$\begin{aligned}
(1 - P) [\int_{s_4}^{s_3} \frac{u + (s_2 - s_1)z_1 + (d - s_2)z_2 - 2d\beta p_o}{t_s * a_j (2 - \zeta) + 2\frac{d}{v}} d(d)] \geq \\
0.9P [\int_{s_1}^{s_2} \frac{u + (d - s_1)z_1 - 2d\beta p_o}{t_s * a_j \zeta + 2\frac{d}{v}} d(d)]
\end{aligned} \quad (25)$$

到现在，尽管我们面对的是单目标优化方程。但是由于本方程中设计的变量过多，对于方程中的两个自变量我们难以求出 Hesse 矩阵，而一阶导数的算取同样有很大困难。故常用的禁忌搜索等算法均不具有适用性。但由于本函数中，两变量范围已经确定，

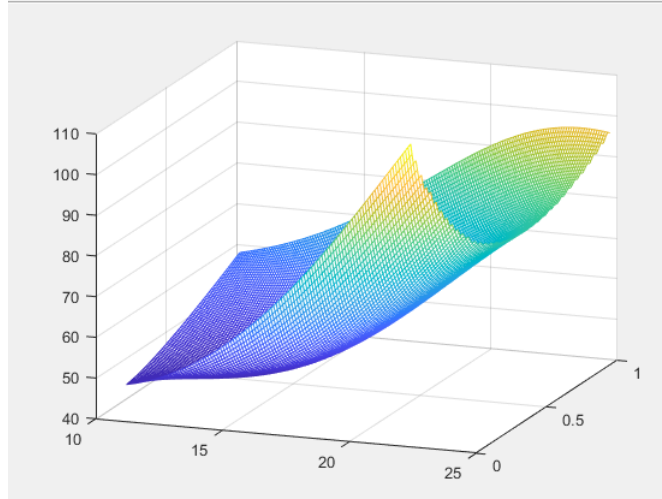


图 11 网格搜索结果 matlab 绘图

较小且有确定的边界。故在此我们选择网格搜索法。我们分别将两个参数的步长设置为 $s(s_4) = 0.1, s(\zeta) = 0.01$, 并对其进行绘图, 其结果如下图所示:

最终在本图中我们得到的最优结果是: $s_4 = 13.11, \zeta = 0.44, D = 42.7498$ 需要注意的是, 在这个模型的实际计算中, 我们选取的是问题一中第一时刻的滞留车数 (即 a_1)。在后续运算中, 我们将问题二中的滞留车辆序列分别输入了本程序中求解, 得到的 s_4 在 12.5-13.5 之间变化, 而 ζ 则在 0.42-0.44 之间变化。可以发现其变化比例相对稳定, 根据前方滞留车辆数的变动情况偏差也不大。故我们近似可以将本题结果作为最优解。

8.2 对按比例给予优先权模型的修正

在本模型的“插队”过程中, 出于简化模型的目的, 我们在令短途车“插队”时, 插入的部分让原本在前方的车辆后退。但是在上面提到的模型中可以直观地从图像中发现, 我们显然面对到的一个问题: 只要在通向载客点的路上始终有车流, 则车辆难以自由转道, 以混合使得模型中所要求的随机混合概率成立。因此在此我们考虑另一种实现方式, 即: 在进入蓄车池过程中分道。而在之前到达入口时领取号牌。根据上一次运载距离安排其号牌种类, 而之后系统自动按照比例 ζ 对进入上车点的车辆顺序进行安排, 以避免在交汇处的混合发生问题。

同时我们需要注意到, 在我们的模型下短途车辆只需要排原车队的 40%-45% 左右的长度, 过大的优先权导致的后果是 1) 司机在路上为了缩短时间导致车速过快, 导致可能的危险; 2) 为了获得短途的优先权采取欺诈的现象。我们通过比例简单比较可以发现, 短途车司机在这种情况下需要排队的时间和数量会让其排队耗时缩短到长途车司机的 $\frac{1}{4}$ 至 $\frac{1}{3}$ 之间。过大的差异难免导致欺诈行为概率的大幅度增加, 而这种欺诈行为将会成为系统中巨大且难以测量、难以量化的负外部性。为了避免欺诈情况的发生、平衡负外部性对于系统的影响, 我们需要对系数 ζ 进行一定的矫正 [6]。经过查阅文献, 我们

认为 ζ 取到 0.6 左右的值是比较合理的，同时对最终方差的影响在可接受的范围之内。

另外，我们建议机场管理部门在实际运营中能通过进一步优化机场候车处的人工管理、通过技术对运载距离进行精准测量等措施对其进行矫正，避免更大的损耗。

九、模型的检验

9.1 对于决策主体“完全理性”假定的检验

本题中预设了一个信息相对透明的情境，即：司机对于航班数量和车辆数量是具有明确认识的，而对于时间、季节等判断则来自于司机的个人经验。在第一问中建立模型时为了简化模型，我们预设本模型中的两个决策主体，即司机和乘客都具有全知的信息。但是在实际情况中进行如此量化可能会对结果产生一定的影响。下面对其进行检验。

由于在本题中，司机对于时间、季节等信息来自经验。因此，不能在某次的单次的决策中将司机视作纯理性。但不可否认的是，来源于经验的信息是可以通过更多的观察和学习进行优化的。因此我们对于这类变量引入“有限理性”的概念 [7], 这种博弈模式能够包含博弈方的学习和策略调整过程, 研究博弈方行为和策略动态稳定性, 从而适用于分析和预测有限理性博弈的分析框架, 是分析能力很差和没有预见能力的个体组成的特定群体内成员间的某种反复博弈。在此我们通过引入调节某些参数产生错误的几率对其过程进行模拟。

具体量化方式可以通过如下方式进行规定：

9.1.1 出租车司机对于季节性因素的学习和策略调整

在本题中，我们在第一、二问的模型中所探究的最终结果是基本没有考虑季节因素的。但其中有两个指标与季节是具有关系的：一个是每天不同时间到达的航班，其随季节变化应当呈现出一定的周期性；二是在探究对乘客选择的影响因素中，其天气部分参数的计算我们直接用“月均降雨率”。在此问题中，前一个信息直接作用于落地的乘客数量，而不论我们是否引入“有限理性”的概念，司机在我们构建的模型中对于“多少飞机落地”的信息都是明晰的，因此唯一需要修正的信息是司机对于单艘飞机中载客量的估计。其可以通过现实数据进一步学习，但在对理性评判中我们不做进一步讨论。

而对于降雨概率的估计，在缺乏经验时会面对偏差，而经过不断地学习和经验其犯错误的概率会降低。因此我们在此将降雨率修正为 (w^* 代表正确预估概率)：

$$w^* \sim N(w, \sigma_1) \quad (26)$$

其代表了司机对天气函数的准确判断具有一定的概率偏高或低，根据 3σ 原则，其最终预估范围在 $w - 3\sigma_1$ 与 $w + 3\sigma_1$ 之间。可以发现越高的 σ_1 代表了司机偏离的范围和预估错误的几率都会变大。

而当我们对其学习因素进行修正，引入学习时间时，我们发现在当前季节 (或月份中)，越长的行车时间 (也代表着越高的学习时间) t_e ，会使其犯错误的几率降低，即其会与 σ_1 反相关。故我们对其进行进一步修正有：

$$w^* \sim N(w, \sigma_1 e^{-\frac{t_e}{T}}) \quad (27)$$

T 代表学习到能够对其评估相对精确的耗时。由此我们对乘客的决策中的季节性因素对天气判断的影响可以进行修正。需注意的是 w 是根据季节 (月份) 变化的外生变量，故乘客学习所修正的认知偏差应当以每次均降雨量变化的时间节点开始重新计数。

9.1.2 司机对于时间因素的学习和策略调整

在问题二中，我们对到达人数情况进行分析；由于其数值比较分散，在高峰时期与平峰时期具有极大的变化，故我们在算法设计中采取输入离散的乘客到达数目，并根据其博弈后的选择，算出每个时间节点中的车辆数，司机再决定是否选择进入蓄车池。

但在此过程中，尽管我们预设司机对乘客数量有充分把握以至于其能够根据前方车辆情况调节是否进入系统。但是不可否认的是，正如我们搜集到的数据显示，高峰时段的航班数量有突然且显著的上漲，在高峰期车辆和乘客数目都显著上升的情况下，司机会对高峰时刻乘客的数目在一定的范围上产生误判。由此在计算时我们进行修正如下：

$$n_0^* = n_0 * (1 + \lambda_1) \quad (28)$$

$$a_j^* = a_j * (1 - \lambda_2) \quad (29)$$

其中， λ_1, λ_2 均在 0-1 之间，代表在高峰时段出租车司机对于飞机数的过高误判和对车辆数的过低误判。

在此考虑学习过程之后，我们对其进行进一步修正：

$$n_0^* = n_0 * (1 + \lambda_1 * e^{\frac{t_s}{T}}) \quad (30)$$

$$a_j^* = a_j * (1 - \lambda_2 * e^{\frac{t_s}{T}}) \quad (31)$$

符号含义与 9.1.1 中相同，但是在涉及到具体参数的修正时要根据具体情况赋值。

9.2 对层次分析法中忽略的“同行人数”参数的检验

在第一题的建模过程中，为了简化计算，我们在公式 (9) 中忽略掉了第三个参数，即同行人数的影响，经过计算我们忽略掉了其低于 4% 的影响权重，只将其对总支出的影

响纳入了考量。在检验过程中，将具体数据即概率分布输入其中产生的结果形式过于繁琐，我们将人数为 1-4 的情况分别纳入考量，直接比较在单个出车口数对人数累积效应的影响，发现在任意时刻，其影响的数值均不超过 25 人，而纳入上车人数的概率分布及成比例计算后，其具体影响效应距离我们所取的 1.28 人的偏差相当小，可以近似视作用 1.28 人代替本因素的影响不存在。故我们认为在上式中对于该因素的忽略合理。

纳入同行人数因素考量后的实验结果如下：

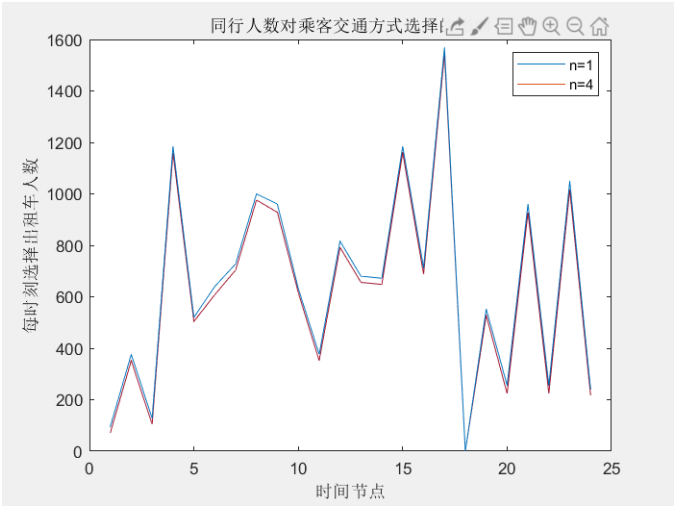


图 12 同行人数对旅客选择乘坐出租车数目情况的影响

十、模型的优缺点

10.1 优点

- 模型建立层次丰富，逻辑连贯，讨论方向比较全面，对于机场出租车问题的整体调度安排模式能够比较完整地拟合。
- 模型中进行的检验较多，在假设框架内，遗留问题和误差较少。
- 将博弈论中的经济学视角和排队论的数理模型有机结合，在其基础上通过层次分析法等建立相对客观的评价体系，得到的结果合乎实际情况，开拓了新的解释视角。

10.2 缺点

- 缺乏实际数据的验证，而从纯数学模型的角度出发；在现实情况中的适用性需要进一步得到证明。
- 在模型的基础上，一些过于复杂的情况中我们对算式进行了简化处理；尽管我们对其中的大部分进行了验证，但是对于部分过于复杂的算式，其解法与准确性还需要进行进一步的验证和优化。

参考文献

- [1] 黄 and 王, “虹桥机场出租车上客系统组织管理优化探讨,” in *2013 城市道桥与防洪第八届全国技术论坛论文集*, (上海), pp. 209–214, 上海市市政公路行业协会, 8 2013.
- [2] . 张 and . 赵, “元胞自动机交通流 NS 模型的参数关联研究,” *昆明理工大学学报 (自然科学版)*, vol. 41, no. 04, pp. 45–51, 2016.
- [3] . 刘, . 王, and . 肖, “单中心城市内部车流密度和行车速度模型研究,” *铁道科学与工程学报*, vol. 8, no. 01, pp. 113–117, 2011.
- [4] . 魏, . 王, and . 邱, “基于排队论的枢纽内出租车上客区服务台优化,” *公路交通科技 (应用技术版)*, vol. 13, no. 10, pp. 298–300, 2017.
- [5] 孙, 基于排队论的航空枢纽陆侧旅客服务资源建模与仿真. PhD thesis, 中国矿业大学 (北京), 2017.
- [6] . 刘 and . 雷, “论外部性理论的内涵及意义,” *西北师大学报 (社会科学版)*, no. 03, pp. 72–75, 2002.
- [7] . 谢, “有限理性条件下的进化博弈理论,” *上海财经大学学报*, no. 05, pp. 3–9, 2001.

附录 1：问题一、二的博弈论、排队论和时间序列的综合模型程序代码

```
1      a=[];
2      a0=1000;
3      b=[1200,1500,1200,2400,1700,1800,1900,2200,2150,1800,1500,2000,
4          1850,1850,2400,1900,2800,1050,1700,1350,2150,1350,2250,1350];
5      b0=20;lastn1=0;
6      w1=0.2548;w2=0.4127;w4=0.2234;w5=0.1091;
7      u=14;z1=2.5;z2=3.6;
8      alpha=0.7;beta=0.056;po=6.66;
9      v=40;s1=3;s2=10;s3=50;
10     v0=30;c0=6;ts=0.0042/15*15;
11     n=1.28;w=0.26;
12     ans=[];
13
14     for k=1:24
15         n0=round(b(k)/8);
16         n1=0;
17         for i=1:n0
18             tp=b0*ts;
19             d=24+3*randn();
20             if d<=s2
21                 c=u+(d-s1)*z1;
22             else
23                 c=u+(s2-s1)*z1+(d-s2)*z2;
24             end
25             C1=w1/(tp+d/v)*d/v0;
26             C2=(w2/c)*n*c0;
27             C3=w4*w/(1-w);
28             C4=w5*1(d/v,0)/1(d/v,n1*8);
29             C=C1+C2+C3+C4;
30             if C<1
31                 n1=n1+1;
32             else
33                 b0=b0+1;
34             end
```



```

35     end
36     lastn1=n1;
37     d=24;
38     while true
39         tw=a0*ts/8;
40         if d<=s2
41             r1=u+(d-s1)*z1-(d)*po*beta;
42             r2=alpha*(u+(v*tw-s1)*z1)-(d+v*tw)*po*beta;
43         else
44             r1=u+(s2-s1)*z1+(d-s2)*z2-(d)*po*beta;
45             r2=alpha*(u+(v*tw-s2)*z2+(s2-s1)*z1)-(d+v*tw)*po*beta;
46         end
47
48         if r1>r2
49             a0=a0+1;
50         else
51             break;
52         end
53     end
54     ans=[ans;k a0 b0*8 n0 n1];
55     if (b0*ts)<0.5
56         a0=a0-round(b0*8/n);b0=0;
57     else
58         tmp=floor(0.5/ts);
59         a0=a0-round(tmp*8/n);b0=b0-tmp;
60     end
61     ans=[ans;k a0 b0*8 n0 n1];
62     end

```

附录 2：模型一的函数 l.m

```

1     function [tmp]=l(t,n1)
2     gamma=0.05;
3     omega=1/640;
4     tmp=(exp(-gamma*t))*(exp(-omega*n1));
5     end

```

附录 3：模型一的检验

```
1  a=[];
2  a0=1000;
3  b=[1200,1500,1200,2400,1700,1800,1900,2200,2150,1800,1500,2000,
4  1850,1850,2400,1900,2800,1050,1700,1350,2150,1350,2250,1350];
5  b0=20;lastn1=0;
6  w1=0.2449;w2=0.3967;w3=0.0387;w4=0.2148;w5=0.1049;
7  u=14;z1=2.5;z2=3.6;
8  alpha=0.7;beta=0.056;po=6.66;
9  v=40;s1=3;s2=10;s3=50;
10 v0=30;c0=6;ts=0.0042/15*15;
11 n=1.28;w=0.26;
12 n3=4;
13 ans=[];
14
15 for k=1:24
16 n0=round(b(k)/8);
17 n1=0;
18 for i=1:n0
19 tp=b0*ts;
20 d=24+3*randn();
21 if d<=s2
22 c=u+(d-s1)*z1;
23 else
24 c=u+(s2-s1)*z1+(d-s2)*z2;
25 end
26 C1=w1/(tp+d/v)*d/v0;
27 C2=(w2/c)*n*c0;
28 C3=w4*w/(1-w);
29 C4=w5*l(d/v,0)/l(d/v,n1*8);
30 CC=w3*n3/4;
31 C=C1+C2+C3+C4+CC;
32 if C<1
33 n1=n1+1;
34 else
```

```

35     b0=b0+1;
36     end
37     end
38     lastn1=n1;
39     d=24;
40     while true
41         tw=a0*ts/8;
42         if d<=s2
43             r1=u+(d-s1)*z1-(d)*po*beta;
44             r2=alpha*(u+(v*tw-s1)*z1)-(d+v*tw)*po*beta;
45         else
46             r1=u+(s2-s1)*z1+(d-s2)*z2-(d)*po*beta;
47             r2=alpha*(u+(v*tw-s2)*z2+(s2-s1)*z1)-(d+v*tw)*po*beta;
48         end
49
50         if r1>r2
51             a0=a0+1;
52         else
53             break;
54         end
55     end
56     ans=[ans;k 8*(n0-n1)];
57     if (b0*ts)<0.5
58         a0=a0-round(b0*8/n);b0=0;
59     else
60         tmp=floor(0.5/ts);
61         a0=a0-round(tmp*8/n);b0=b0-tmp;
62     end
63     end

```

附录 4：矩阵式车道模型代码

```
1      ts=0.0042;v1=5;tg=0.0042/15*6.48;lc=4.7/1000;d=0.3/1000;
2
3      x=1:4/100:5;K=[];
4      yy=[];
5      for i=1:101
6          v=20*(1-(2*x(i)-1)/(50/(5+0.5)));
7          yy=[yy v];
8          K=[K,2*(2*x(i)-1)/
9              ((ts+tg+(2*x(i)-1)*(lc+d)/v)+(x(i)-1)*(lc+d)/v1)];
10         end
11
12         x=x*2-1;
13         figure(1);
14         plot(x,K);
15         title('乘车效率与上车点数目关系');
16         xlabel('上车点数目');
17         ylabel('乘车效率');
18         saveas(1,'matlabchart3.png');
19
20         figure(2);
21         plot(x,yy);
22         title('行车平均速度与上车点数目关系');
23         xlabel('上车点数目');
24         ylabel('行车平均速度');
25         saveas(2,'matlabchart4.png');
```

附录 5: 交织车道模型代码 (对 d 和 s_w 关系的分析)

```
1      lc=4.7;ts=0.0042/15*20;v1=5;v2=20;
2
3      [sw,d]=meshgrid(1:0.5:20,4:1:27);
4      n=6;
5
6      K=[];
7      ans1=0;
8      for i=1:39
9          for j=1:24
10             if (2.42*(d(j,1)+lc)^0.234)/(sw(1,i)^0.438)<=1.4
11                 K(j,i)=n/(ts+((n*lc+n*d(j,1)-d(j,1))/1000)/(4*v1)+
12                     (2*(lc+d(j,1))/1000)/sw(1,i)+((n-2)*lc+(n-3)*d(j,1))/1000/v2);
13                 if ans1<K(j,i)
14                     ans1=K(j,i);
15                     jj=j;ii=i;
16                 end
17             else
18                 K(j,i)=0;
19             end
20         end
21     end
22
23     surf(sw,d,K);
24     [ans1 jj ii]
```

附录 6：交织车道模型代码 (对 n 与 d 、 s_w 关系的分析)

```
1      lc=4.7;ts=0.0042/15*15;v1=5;v2=20;
2
3      [sw,d]=meshgrid(1:0.5:20,4:1:27);
4      KK=[];
5
6      for n=3:6
7          K=[];
8          ans1=0;
9          for i=1:39
10             for j=1:24
11                 if (2.42*(d(j,1)+lc)^0.234)/(sw(1,i)^0.438)<=1.4
12                     K(j,i)=n/(ts+((n*lc+n*d(j,1)-d(j,1))/1000)/(4*v1)+
13                         (2*(lc+d(j,1))/1000)/sw(1,i)+((n-2)*lc+(n-3)*d(j,1))/1000/v2);
14                     if ans1<K(j,i)
15                         ans1=K(j,i);
16                         jj=j;ii=i;
17                     end
18                 else
19                     K(j,i)=0;
20                 end
21             end
22         end
23         KK=[KK ans1];
24     end
25     plot(3:6,KK);
```

附录 7：插队模型的网格搜索代码

```

1      s1=3;s2=10;u=14;z1=2.5;beta=0.056;po=6.66;z2=3.6;
2      ts=0.0042;v=40;s3=45;
3      a=[1687 1587 1781 1143 1293 1387 1318 1143 1143 1356 1593 1249
4      1356 1356 1143 1149 1143 1406 1456 1687 1143 1681 1143 1624];
5      a=a./8;
6      j=1;
7      [S4,ZETA]=meshgrid(10.1:0.1:40,0.2:0.01:1);Z=zeros(81,300);
8      ZZ=zeros(81,300);
9      x=11:40;y=[];minD=99999999;I=0;J=0;
10     for i=1:300
11     for jj=1:81
12     s4=S4(1,i);zeta=ZETA(jj,1);
13     ss4=3:0.1:s4;
14     P=trapz(ss4,exp(-((ss4-24).^2./98))./(7*sqrt(2*pi)))/0.9973;
15
16     d=s1:0.02:s2;
17     fenmu=(ts*a(j)*zeta+2*d./v);
18     t1=P*trapz(d,(u+(d-s1).*z1-2*beta*po*d)./fenmu);
19     d=s2:0.02:s4;
20     fenmu=(ts*a(j)*zeta+2*d./v);
21     t2=trapz(d,(u+(s2-s1).*z1+(d-s2).*z2-2*d*beta*po)./fenmu);
22     d=s4:0.02:s3;
23     fenmu=(ts*a(j)*(2-zeta)+2*d./v);
24     t3=(1-P)*trapz(d,(u+(s2-s1)*z1+(d-s2).*z2-2*d*beta*po)./fenmu);
25     if (t3/t2<0.9)|| (t3/t1<0.9)
26     Z(jj,i)=nan;ZZ(jj,i)=nan;
27     continue;
28     end
29     E=(t1+t2+t3)/(s3-s1);
30
31     d=s1:0.02:s2;
32     fenmu=(ts*a(j)*zeta+2*d./v);
33     t1=P*trapz(d,((u+(d-s1).*z1-2*beta*po*d)./fenmu-E).^2);
34     d=s2:0.02:s4;

```

```

35     fenmu=(ts*a(j)*zeta+2*d./v);
36     t2=trapz(d,((u+(s2-s1).*z1+(d-s2).*z2-2*d*beta*po)./fenmu-E).^2);
37     d=s4:0.02:s3;
38     fenmu=(ts*a(j)*(2-zeta)+2*d./v);
39     t3=(1-P)*
40     trapz(d,((u+(s2-s1)*z1+(d-s2).*z2-2*d*beta*po)./fenmu-E).^2);
41     D=(t1+t2+t3)/(s3-s1);Z(jj,i)=D;ZZ(jj,i)=E;
42     if minD>D
43         I=s4;J=zeta;
44         minD=D;
45     end
46
47 end
48 end
49 mesh(S4,ZETA,Z);
50 figure(2);
51 mesh(S4,ZETA,ZZ);
52
53 [I,J,minD]

```