

小区开放问题研究

摘要

现代社会逐步走向开放、多元化，人们的居住环境也逐步走向开放化；同时，随着经济的高速发展，各个城市的交通压力不断增大，交通问题正逐渐引起人们的关注。本文旨在找到合适的目标函数来衡量小区周围的道路通行能力，再建立小区开放对各指标的影响模型，进而研究不同情况下小区开放对目标函数的影响情况。在此基础上，考虑在对小区周围道路通行能力大幅提升后，交通管理部门应当采取何种开放方式使得综合效用达到最大。

对于问题一，通过查阅文献和参考生活经验列出可能会影响小区周围道路通行能力的指标，分别从**车辆、时间、道路、效用**四个方面出发选取合适的指标，基于各指标分别建立了**路网交通效率、拓展 MSA 模型和综合效用**三个单目标模型，利用**集结算子**对三个单目标模型进行加权最终得到**多目标规划**问题来综合衡量小区周围道路的通行能力。

对于问题二，小区以某种方式开放会影响问题一中各指标的取值，进而影响路网交通效率及 MSA 模型中的阻抗，在只考虑交通通行能力的情况下，本文将综合效用作为常数考虑。文章以单调性关系在三个单目标模型上建立了一个比率系数，从而得到小区以具体形式开放后的三个单项目标新取值。以**二元语义**的形式采集了三个指标的重要性程度，用**模糊层次分析法**得到了三个单目标函数的权重系数，最终采用问题一的加权模型解释了小区以具体开放形式对道路通行能力的影响。并且分析了不同的开放形式对各单目标函数的影响，利用数据模拟，得出小区开放对缓解交通压力有一定的帮助的结论。

对于问题三，基于小区周围的车流量等历史数据建立了 **ARIMA**，**灰色预测**，**神经网络** 三个单项预测模型，建立了一个基于 **IOWA 算子**的**组合预测**的模型用来预测某个小区在未来的几年内的交通压力情况，从而根据有无交通压力**对小区进行分类**。在分类的基础上，利用问题二中建立的模型，将小区分成5种结构，通过对车流量、小区面积、外部道路通行能力等指标的变化来定量比较不同类型小区开放后对道路通行的影响，得到了交通压力较大的小区开放后能大幅度增加道路通行能力，在交通压力较小的小区开放后对道路通行能力的增加不够明显，同时小区结构，小区面积，周边道路情况等众多因素也对道路通行能力提高有着重要的影响。

对于问题四，将路网交通效率，MSA 模型中的道路阻抗，综合效用作为属性值，不同的开放方式和修路方式作为方案，建立了一个**基于 TOPSIS 方法的多属性决策模型**。通过**定量**分析来选择最适合的开放方案，再从**定性**分析的角度向相关部门提出一系列关于小区开放的合理化建议。

关键词：路网交通效率；拓展 MSA 模型；模糊层次分析法；组合预测；TOPSIS 方法

一、问题重述

随着经济的高速发展，城市对道路的通行能力提出了更高的要求，但是封闭式的小区 and 单位大院却阻碍了城市路网结构的完善，降低了道路的可达性。国务院推广的街区制中，提出已建成的住宅小区和单位大院要逐步开放，以后不再建设封闭住宅小区的意见。开放小区后对周边道路通行产生的影响该怎样评价以及是否优化了路网结构等问题引起了广泛讨论。

小区开放政策是为了优化道路通行能力，本文研究了以何种评价指标体系来衡量道路交通的状况，关于车辆通行要建立怎样的数学模型，论证如何才能合理有效化地开放小区并就此向有关部门提出合理化的建议，具有一定的理论价值和实际意义。

本文需要解决以下问题：

- 1、建立合适的评价指标体系，评价小区开放对周边道路通行的影响。
- 2、建立关于车辆通行的数学模型，以研究小区开放对周边道路通行能力的影响。
- 3、利用已建立的模型，构建具体的函数，定量分析不同类型小区开放前后对道路通行能力。
- 4、向城市规划部门和交通管理部门提出关于小区开放的合理化建议。

二、问题分析

问题一的分析

在问题一中，我们需要解决的问题是如何选择合适的评价道路通行能力的指标，然而由于影响道路交通情况的因素复杂多样，评价指标也相应地变得非常复杂繁多。因为指标较多，为了简化分析，我们对指标进行了粗糙简约的属性集中方法。

本题中，我们将所选择的指标归为道路指标、时间指标、车辆指标、小区指标、效用指标、其他指标五大类。为了形成精简高效的评价体系，对前四大类指标进行了进一步的综合和量化。总共建立了交通效率模型、*MSA* 模型和综合效用模型，最后根据这三个模型又建立起了能综合评价道路通行能力的多目标模型，得到最终的评价体系，作为全文的宏观模型。

问题二的分析

对于问题二，我们要研究小区开放对周边道路的影响，首先要做的是对前一问中建立的三个模型的重要性程度进行确定，对此可利用二元语义确定各目标的权重。之后可再建立小区开放对各指标影响的模型，修正各个指标，得到相应新的函数指标，再利用前一问中的多目标模型，得到对小区周边通行能力的新的评价模型，进而研究影响关系。

问题三的分析

问题三需要考虑到小区周边未来的交通压力是影响小区开放的重要因素，而交通压力可由车流量来衡量。因此可以采用基于 *IOWA* 算子的组合预测模型法，并对灰色预测、神经

网络预测、*ARIMA* 预测三个单项预测结果进行综合，得到未来各小区周围的交通压力。随后在问题二的基础上，构建不同类型的五种小区，构建基于 *BPR* 函数的拓展 *MSA* 模型，用以确定小区开放后道路的阻抗函数，再利用 *MATLAB* 软件计算不同小区模型、不同车流量、不同道路结构下小区开放后的车辆通行能力，比较不同情况下开放后的车辆通行能力。这样就可得到小区结构、周边道路结构、车流量对于小区开放效果的影响。

问题四的分析

在问题四中我们要考虑的是不同情况下小区是否该开放，以及应该如何进行开放的政策选择问题。此时首先建立基于 *TOPSIS* 方法的多属性决策模型，定量分析小区应该选择何种方案。再从定性分析的角度提出关于小区开放的一系列建议。

三、模型假设

- 假设一：收集到的数据都是真实可靠的
- 假设二：在对模型的分析中路面情况是理想的，不会发生交通事故、车道被占用等情况
- 假设三：所有的司机都是理智的并且不会违反交通规则
- 假设四：用模糊层次分析法得到的二元语义矩阵具有代表性
- 假设五：模型的细微因子当作常量并对模型的造成的误差在可接受的范围以内
- 假设六：小区内车速保持在合理范围内

四、符号说明

c	道路通行能力
q	车流量
v	服务交通量
V	车速
v/c	饱和度，指服务交通量与基本通行能力的比值
T	一个信号灯周期
f_1	衡量路网交通效率的函数
f_2	衡量车辆通过效率的函数

f_3	衡量车辆通过效率的函数
t_2	行程时间
U_1	司机效用函数
U_2	居民效用函数

五、模型的建立与求解

5.1 问题 1 的建立与求解

5.1.1 基本分析

2016 年 2 月 21 日，国务院发和谐布《关于进一步加强城市规划建设管理工作的若干意见》，其中第十六条关于推广街区制，原则上不再建设封闭住宅小区。这对中国的城市建设具有里程碑般的意义，因此开放包容是一个现代化城市的标志，也意味着社会走向更加开放，更加多元的未来。开放型小区可以畅通城市毛细血管，健全城市的道路网络。但是一些交叉路口的过往车辆可能也会阻碍城市的通车能力。开放型小区对道路通行能力的影响引起了公众的争议，因此建立一个小区开放对周边道路通行的影响的评价体系尤为重要。我们从四大方面建立了指标体系，分别是道路指标、时间指标、车辆指标和效果指标。

5.1.2 影响因素分析

影响小区开放产生效果的因素有小区位置、小区结构、小区面积和小区内行人密度。

小区位置 a 指距离市中心的距离距离，单位为千米 (km)；小区构造 w 按内部结构可分为的片块式布局小区、轴线式布局小区、向心式布局等；行人密度 ρ 指小区单位面积内行人的数量，单位为人/平方千米($人/km^2$)。

不同的小区反映着不同的交通情况，是考虑缓解交通压力的具体情况。

5.1.3 评价指标体系的构建

(1) 道路指标

道路指标包括路网密度、饱和度、排队长度和交叉路口多少。

其中路网密度 p 指所有的道路的总长度 l_2 与小区的面积 s 之比，即 $p = l_2 / s$ ，单位为米/每平方米 (m/m^2)；

饱和度 v/c 指服务交通量与基本通行能力的比值，体现的是一种道路服务水平。服务交通量 v 与实际通行能力 c 的比值。当 v/c 的比值逐渐增长，行车速度呈下降趋势。具体表现形式为：

$$\begin{cases} v/c < 0.6, & level - 1 \\ 0.6 \leq v/c < 0.75 & level - 2 \\ 0.75 \leq v/c < 0.90 & level - 3 \\ 0.90 \leq v/c < 1.00 & level - 4 \\ v/c > 1.0 & level > 4 \end{cases}$$

(其中 $level$ 表示道路服务水平的级数);

排队长度 l_1 指通过路口排队长度的平均值, 单位为米 (m);

交叉路口多少 n 指单位公里数内交叉路口的多少。

结合实际情况可知, 路网密度反映了小区范围内的主干路、次干路和支路以及小区内的道路构成的网络体系结构, 其能从整体上反映小区周围的交通情况, 同时合理的交叉路口的设计也能在适当的位置对车辆进行分流, 从而能反映小区周围的交通压力。

(2) 时间指标

时间指标包括延误时间和行程时间。

延误时间 t_1 指车辆通过交叉路口时等待红绿灯的时间, 单位为秒 (s);

行程时间 t_2 指车辆通过单位公里的平均行程时间, 表达式为 $t = s/v$, 单位为秒 (s)。

延误时间和行程时间是反映通行压力的直观指标, 延误时间和行程时间越短, 则可反映出小区周围的交通压力越小。

(3) 车辆指标

车辆指标包括车流量、车速和过往车辆种类。

车流量 P 指的是单位时间通过的车辆数目, 若 v 为平均速度, t 为单位时间;

l 为道路通畅时平均车距, L 为车身长, 通过一段距离为 s 的路段车数量为 n , 则 $P = t * v / (l + L)$; 车速 v 指所有车辆在每个小路段当中的平均速度, 单位为米/每秒 (m/s);

车辆种类分为四类, 其中自行车为 1, 摩托车为 2, 客车为 3, 货车为 4。

不同的车辆会对交通造成直接影响, 自行车、摩托车和行人等行驶方向的不确定性会干扰交通的正常运行, 同时不同车辆的类型会在道路和视线上影响绝对交通情况的判断, 进而影响小区周围的交通压力。

(4) 效用指标

效果指标包括司机效用, 居民效用和交通部门效用。

司机效用 U_1 主要是受到行车距离和单位距离行车时间影响:

$$U_1 = \phi T(t) + \alpha D_1 - C_1 \quad (1)$$

其中 $T(t)$ 为单位距离行车时间, 其由行车时间和延误时间综合决定, D_d 为司机穿过该小区到达目的地的实际行驶距离, 应该根据道路开通前后行驶路线的不一样来分别考虑, C_d 为司机群体的异质性, 在区间 $[0, C_1]$ 上服从均匀分布。

居民效用函数 U_2 主要受到开通小区后居民的交通更为便利的影响, 同时小区开通后会

有噪音问题和安全问题会从反方向对效用函数造成影响,交通便利优化程序可由下文设计的交通能力综合评价指标函数 f , 噪音和安全问题可根据实际的车流量进行分级, 具体体现为:

$$U_2 = f + \alpha_2 - C_2 \quad (2)$$

其中 C_2 反映具名的异质性问题, 在区间 $[0, C_2]$ 上服从均匀分布, 用来综合反应居民的平均偏好水平, 包括偏好于交通更为便利和更安静安全等。

交通部门效用函数 U_3 主要体现在两个方面, 一方面为开通小区后对小区周围的交通压力缓解情况, 另一部分为开通小区的费用成本。

(5) 其他指标

其他指标包括车道面积, 路网密度, 车道数目, 道路类型。

车道面积 s 指某个区域内道路的总面积与该区域的面积之比, 单位为平方米(m^2)。

车道数量 n 指开放小区内车道的总数量。

路网密度 p 指所有的道路的总长度 l_2 与小区的面积 s 之比, 即 $p = l_2 / s$, 单位为米/每平方米, 即(m / m^2);

道路类型 k 具体指快速路, 主干路, 次干路, 城市支路

这些指标虽然在三个模型中没有具体体现, 但仍是周边道路影响的重要因子。由于所起作用与指标体系中的某些评价指标有所重复, 或是若考虑该指标会使模型过于特殊化, 不适用于各小区一般普遍情况, 实际应用意义不强, 为了模型的精练性和普适性, 在模型的建立中我们并没有考虑这些指标。

5.1.4 整体体系思路

构建不同小区开放效果的评价指标的整体思路如下所示:

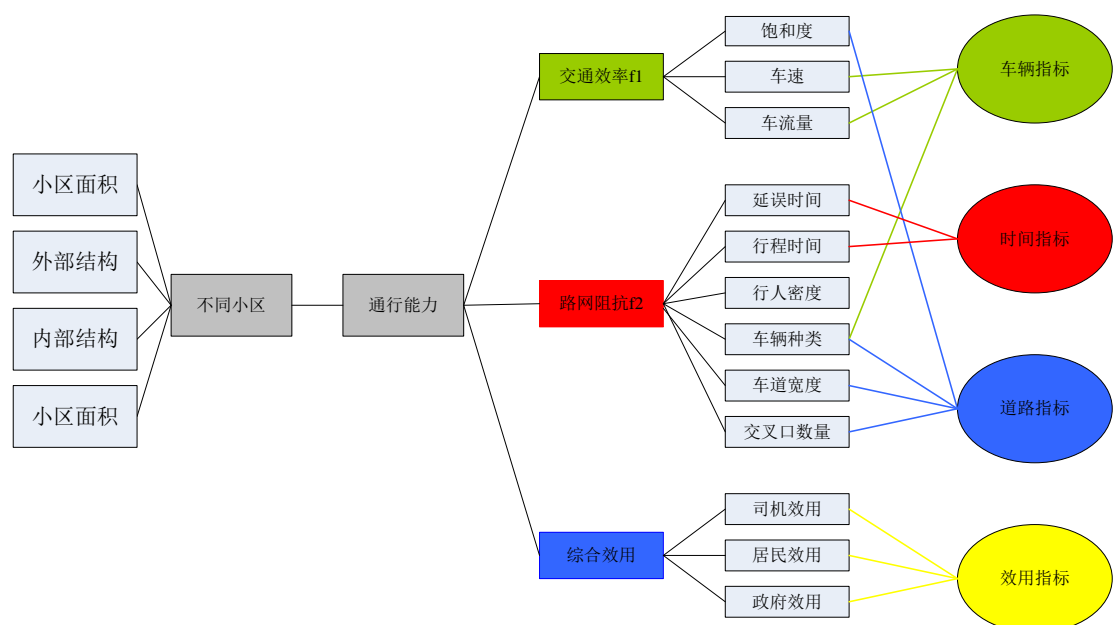


图1 小区开放对道路影响的评价指标体系

5.1.3 通行能力综合评价模型的构建^[8]

(1) 交通效率模型

交通效率模型主要应用通行能力 c 车流量 q 车速 v 饱和 q/c 等指标,由单个路口的交通效率、单条路径的交通效率,进而逐步推广至多个小区周围道路的路网交通效率,具体步骤如下:

Step1: 单个路口的交通效率

单个路口 a 的交通效率 σ_a 体现为该路段的饱和度与其交通流量饱和度的乘积,表示路段达到一定车辆转移需求量的同时,要避免通过该路口的车辆滞慢程度增大,其函数关系为:

$$\sigma_a = \frac{q_a}{c_a} \times \frac{v_{a_1}}{v_{a_2}} \quad (3)$$

(其中 q_a 为路段 a 的车流量, c_a 为路段 a 的通行能力, v_{a_1} 为路段 a 的实际车速, v_{a_2} 为路段 a 的自由车速)。

Step 2: 单条路径的交通效率

由于交通周转量总和 $f_k(ql)$ 是直接反映某条路径的交通效率的因素,为了计算某个路段 x 交通效率,我们首先考虑该条路径的交通周转量总和:

$$f_k(ql) = \sum_{i=1}^n q_i l_i \quad (4)$$

(其中 q_i 为第 x 条路径的第 i 条路段的车流量, l_i 为第 x 条路径的第 i 条路段的长度, n 为第 x 条路径的路段总数)。

根据各路段的交通周转量不同,其对路径的交通效率的影响程度也不同,则可知第 x 条路径的交通效率为:

$$\sigma'_x = \sum_{i=1}^n \sigma_i \frac{q_i l_i}{f_k(ql)} \quad (5)$$

Step 3: 路网的交通效率

小区周围的道路网,其实际为不同的道路交叉,具体体现为多个路口和多条路径的结合,根据交通路网负载均衡原理,交通流量最大的路径其出行时间是最小,即最小时间最大流路径,记其效率为 $\sigma_{x-\min-\max}$,根据每条路径需求量的大小可确定各条路径所占的权重 w ,则路网交通效率 δ 为:

$$\begin{aligned} \delta &= \sum_x \sigma_{x-\min-\max} \times w_x \\ &= \sum_x \left[\sum_{i=1}^m \sigma_i \frac{q_i l_i}{f_k(ql)} \right]_{\min-\max} \times \frac{f_{x-\min-\max}(ql)}{\sum_x f_{x-\min-\max}(ql)} \end{aligned} \quad (6)$$

(2) 改进 MSA 模型建立求解

在 MSA 模型中, 我们建立小区的结构, 为了简化模型, 我们只考虑从 A 到 C 的阻抗 (行程时间), 用有向图表示表示行车方向, 同时将各种车型均换算成标准车辆, 假设小区的形状是对称的, 则 A 到 C 的通行状况可以用来替代多种通行状况。该模型中要求模拟小区周边以及内部基本数据, 包括周边道路通行能力, 交叉口水平, 小区内道路通行能力, 行人自行车密度, 车流量饱和度, 车速, 小区内道路结构等。通过计算得到了整个路网的阻抗函数。

Step 1 路段阻抗函数的修正 BPR 函数

小路段实际阻抗函数为:

$$t^{i,j} = t_0^{i,j} (1 + \alpha (\frac{x_{i,j}}{C_{i,j}})^\beta) \quad (7)$$

(其中 $t_0^{i,j}$ 是路段 I, J 上自由流时的走行时间; $C_{i,j}$ 是路段 I, J 的设计通行能力; $x_{i,j}$ 是路段 I, J 的实际车流量)。

小区的支路要考虑到自行车和行人的影响, 对 $C_{i,j}$ 进行修正, 即建立修正的 BPR 模型:

$$t^{i,j} = \begin{cases} t_0^{i,j} (1 + \alpha (\frac{x_{i,j}}{\eta \eta_1 C_{i,j}})^\beta), 0 \leq \eta_1 \leq 1 \\ t_0^{i,j} (1 + \alpha (\frac{\eta_1 x_{i,j}}{\eta C_{i,j}})^\beta), \eta_1 \geq 1 \end{cases} \quad (8)$$

(对于 α, β 的计算, 可取回归结果; η 为行人的干扰系数; η_1 为自行车干扰系数。)

Step 2 交叉路口阻抗函数

针对不同的交叉路口类型以及车辆在路口处不同的运动状态, 采用不同的计算参数和计算公式从而得到各个路口各种运动状态的阻抗函数 t 。

1. 主路和支路的交叉口

分别计算从主路右转进入支路, 在主路直行, 从支路左转进入主路三种情况:

右转情况我们考虑右转专用道的阻抗:

$$t_{right} \approx 3 \sim 3.6s$$

左转情况考虑直行和左转车道分离, 计算左转的阻抗:

$$t_{left} = T_p / n_{left} \quad (9)$$

(其中, T_p 是一个信号灯周期 n_{left} 是一个周期允许左转的车辆数)。

同理直行情况考虑直行的绿灯时间较长, 直行和左转车道分离。

$$t_{stra} = T_p / n_{stra} \quad (10)$$

2. 支路和支路的交叉口

对于支路同样考虑车辆的直行, 右转, 左转三种运动状态。但是小区内道路一般比较狭窄, 会存在直行和转向的车辆混行的状态, 对公式进行拓展修正, 引入折减系数, 具体公式为:

$$t'_{left} = t_{left} (1 - \frac{n_l}{2} K) \quad t'_{right} = t_{right} (1 - \frac{n_r}{2} K) \quad t_{stra} = t_{stra} - \frac{n_s}{2} K \quad (11)$$

3.主路和主路的交叉路口

我们也要考虑车流转向的阻抗,利用相似的计算方法得到主路和主路交叉路口各种运动状态的阻抗函数。

Step 3 基于通行路线的搜索的拓展 MSA 模型

对于给定的小区开放结构,我们首先要搜索找寻可能存在的所有路线。由于正向行驶和逆向行驶考虑方法一样,本文仅针对正向车流考虑,例如从 A 到 C 的运行。交叉路口和交叉路口之间的小路段构成了有向图。据此建立有向图路径搜寻的数学模型。我们要寻找 A 到 C 的路线,建立有向图深度优先搜索的递归算法。如下图我们要寻找顶点 v_0 到 v_5 的所有可能路线,其基本思路是,找到和 v_0 直接相连的节点, v_1 和 v_2 , 再分别找到 v_1 到 v_5 的所有可能路线和 v_2 到 v_5 的所有可能路线,由此我们一步步的向下递归,直至起点与终点重合。

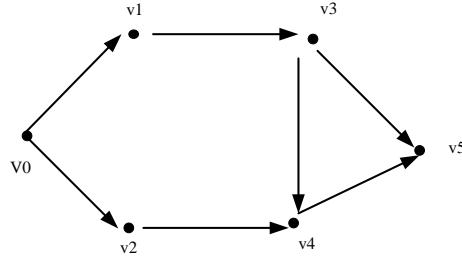


图 2 深度优先递归两点间路线示意图

由上可以得到任意一个小区结构的所有可能路线。针对任一条路线,都是有若干个小路段和交叉口组成。则有任意路线的阻抗函数表达式:

$$T_{path,i} = \sum_{j=1}^m t_r^j + \sum_{k=1}^l t_c^k$$

$$t_r^j = t_0^{p,q} (1 + \alpha (\frac{x_{p,q}}{C_{p,q}})^\beta) \quad (12)$$

$$t_c^k = \begin{cases} t_{left}^k \\ t_{stra}^k \\ t_{right}^k \end{cases}$$

(其中, $T_{path,i}$ 是第 I 条路线的阻抗函数; t_r^j 是第 I 条路线上第 J 小路段的阻抗函数, t_c^k 是第 I 条路线第 k 交叉路口的阻抗函数)。

给定路网信息。路线信息,对于特定的交通量,我们需要计算交通量在整个路网的分布。根据实际情况,车流量的分布应该趋于一种平衡的状态使得各条路线的阻抗基本相等。这样我们才有容量限制—增量加载的算法,迭代的计算整个路网的车流量分布。从而根据整个路网的车流量分布计算出整个交通网的总阻抗,其计算公式为:

$$\begin{aligned}
T &= \sum_{i=1}^n T_{path,i} \times V_{path,i} \\
&= \sum_{i=1}^n \left(\sum_{j=1}^m t_0^{p,q} (1 + \alpha (\frac{x_{p,q}}{C_{p,q}})^\beta) + \sum_{k=1}^l t_c^k \right) \times V_{path,i}
\end{aligned} \tag{13}$$

(3) 综合效用模型

综合效用 U 为司机效用居民效用和交通管理部门三个方面效用的综合, 根据式(1)(2)

$$\begin{aligned}
U &= U_1 + U_2 + U_3 \\
&= \phi T(t) + \alpha D_1 - C_1 + f + \alpha_2 - C_2 - kC
\end{aligned} \tag{14}$$

(其中 C 为在考虑了 U_1, U_2 效用的基础上增加的成本指标, k 为比例系数)。

(4) 综合评价道路通行能力的多目标模型

由路网交通效率、改进 msa 模型和综合效用三个单目标模型加权得到如下多目标模型:

$$\begin{aligned}
f &= w_1 f_1 + w_2 f_2 + w_3 f_3 \\
&= w_1 \left(\sum_x \left[\sum_{i=1}^m \sigma_i \frac{q_i l_i}{f_k} (ql) \right]_{\min-\max} \times \frac{f_{x-\min-\max}(ql)}{\sum_x f_{x-\min-\max}(ql)} \right) \\
&\quad + w_2 \left(\sum_{i=1}^n \left(\sum_{j=1}^m t_0^{p,q} (1 + \alpha (\frac{x_{p,q}}{C_{p,q}})^\beta) + \sum_{k=1}^l t_c^k \right) \times V_{path,i} \right) \\
&\quad + w_3 (\phi T(t) + \alpha D_1 - C_1 + f + \alpha_2 - C_2 - kC)
\end{aligned} \tag{15}$$

(其中 $w = (w_1, w_2, w_3)^T$ 为路网交通效率, 交通网络总阻抗和综合效用的重要性程度)。

本文将该模型作为全文的宏观模型, 用来综合评价道路通行能力, 主要考虑了后续小区开放后对各因子的影响, 进而影响各单目标, 最终影响综合评价指标, 在不考虑综合效用的时候, 本文将 f_3 设定为常数。

5.2 问题 2 模型的构建与求解

5.2.1 基本分析

问题 2 是通过建立模型分析小区开放对周边道路交通能力的影响, 首先利用基于模糊层次法的二元语义对各目标的重要性程度进行确定。在此基础上建立小区开放对各指标影响模型, 进一步修正各个指标, 然后将新的指标带入问题一中的多目标模型, 由此得到对小区周边通行能力的新的评价模型。

5.2.2 模型的建立与求解

Step 1: 各目标模型重要性程度的确定

通过权重的确定, 可以体现指标的重要性程度。因此首先我们引入二元语义的概念: 设 S 是预先定义好的由奇数个元素 $l+1$ 构成的有序自然语言评价值, 即 $S = (s_0, s_1, \dots, s_l)$,

假设 $(s_i, a_i), (s_j, a_j)$ 为两个二元语义，其中 a_i, a_j 为符号转换值，表示由计算得到的语言信息与预先定义的语言信息集 S 中最贴近语言短语之间的偏差，很显然它满足 $a_i, a_j \in [-0.5, 0.5]$ 。

于是我们得到：

- 1) $s_i > s_j \Rightarrow (s_i, a_i) > (s_j, a_j)$;
- 2) $s_i = s_j \text{ and } a_i = a_j \Rightarrow (s_i, a_i)$;
- 3) $s_i = s_j \text{ and } a_i > a_j \Rightarrow (s_i, a_i) > (s_j, a_j)$;

假设 $S = (s_0 \ s_1 \ s_2 \ s_3 \ s_4)$ ，其中 $s_0 = 0$ ， $s_l = 1$ ，则可建立如下的转化关系，将实数 $\beta \in [0, 1]$ 转化为二元语义：

$$\Delta(\beta) = (s_i, a_i) \begin{cases} i = \text{round}(\beta l) \\ a_i = \beta - \frac{i}{l} \in \left(-\frac{1}{2l}, \frac{1}{2l}\right) \end{cases} \quad (16)$$

同样也可以用下面的公式将 S 中的二元语义转化为区间 $[0, 1]$ 的数 β ：

$$\Delta^{-1}((s_i, a_i)) = \beta = \frac{i}{l} + a_i \quad (17)$$

道路通行能力的主要评价因素在于某一路段单位时间通过的车辆数目，对于不同的小区，各个指标的重要程度不尽相同，对于周边交通压力较低的小区，开放小区道路对缓解交通压力的效果并不是很明显，而交通压力大的小区，车流量是最重要的因素，对于面积较大的小区，为了节省行驶距离，通过小区的行程时间又显得更加重要些。然而对两个指标之间的评价我们只能给出类似于重要，次要，同等重要的语言刻度，为了使得指标的重要性程度更具有代表性，我们对不同的小区采取实地调查的方式得到二元语义权重矩阵，我们将道路通行能力指标重要性程度分成 5 个等级，用语言集表示为：

$$S = (s_0 \ s_1 \ s_2 \ s_3 \ s_4) = (\text{很不重要} \ \text{不重要} \ \text{同等重要} \ \text{重要} \ \text{很重要})$$

我们将所得到的调查数据进行统计后，所得到的二元语义矩阵分类如下：

$$\begin{aligned} \text{第一类: } & \begin{pmatrix} (s_2, 0) & (s_1, 0) & (s_3, 0) \\ (s_3, 0) & (s_2, 0) & (s_4, 0) \\ (s_1, 0) & (s_0, 0) & (s_2, 0) \end{pmatrix}, \text{ 所占比例大致为 } 80\%, \\ \text{第二类: } & \begin{pmatrix} (s_2, 0) & (s_1, 0) & (s_3, 0) \\ (s_4, 0) & (s_2, 0) & (s_4, 0) \\ (s_0, 0) & (s_1, 0) & (s_2, 0) \end{pmatrix}, \text{ 所占比例大约为 } 20\%, \end{aligned}$$

运用二元语义逆算子的公式 (17) 将所得到的重要性语言指标进行集结得到实数矩阵：

$$P = \begin{pmatrix} 0.500 & 0.250 & 0.750 \\ 0.800 & 0.500 & 1.000 \\ 0.200 & 0.050 & 0.600 \end{pmatrix}$$

求得权重向量 $W = (0.3309 \quad 0.5367 \quad 0.1324)$ 且通过 $n=3$ 时的一致性检验。

Step 2: 小区开放对各指标的影响模型构建^[7]

小区开放以后，对路网交通效率、道路延误时间和综合效用都在不同的程度上起到一定的影响，考虑到小区开放主要体现在小区开通方式和开通结构，为了简化计算，我们将其概括为开通道路数目 Rn 和交叉口数目 Cn 对各指标的影响，而各指标又会以一定的单调性关系进而影响路网交通效率、道路延误时间和综合效用，因此以相应的正负比例关系建立三个关于开放小区后的三个新的目标函数，改进方式如下：

新的路网交通效率：

$$f_1' = \left(\frac{Rn}{(Rn+Cn)} \right) \times f_1 \quad (18)$$

从新指标我们可以看出，小区开放后，小区的道路上数目的增加会提升路网的交通效率，但是交叉口数目的过多同时在一定程度上会制约路网交通效率的大幅度提高。

新的道路延误时间：

$$f_2 = \left(1 / \left(\frac{Rn}{Rn+Cn} + 1 \right) \right) \times f_2 \quad (19)$$

从新指标我们可以看出，整体来看小区开放后会在一定的程度降低道路延误时间，同时体现了如果交叉口的增多在一定程度上会增加路口的等待时间，在某个程度上会对道路延误时间造成负面影响。

新的综合效用：

$$f_3' = \ln \left(\left(\frac{Rn}{Rn+Cn} + 1 \right) \right) \times f_3 \quad (20)$$

由上述函数可知，小区开放，随着小区通车道路数的增加和路口数目的增加，会对综合效用产生一定的积极影响，考虑到效用不会无限增大，本文以对数形式加以考虑。

Step 3: 小区开放道路通行能力评价模型的构建

利用 step2 中的三个新指标函数，根据式 (15)，代入到本文在问题 1 中所建立的宏观模型，建立了如下小区开放后对道路通行能力的新评价模型：

$$\begin{aligned}
f &= \left(\frac{Rn}{(Rn+Cn)} \right) f_1 + \left(1 / \left(\frac{Rn}{Rn+Cn} + 1 \right) \right) f_2 + \ln \left(\left(\frac{Rn}{Rn+Cn} + 1 \right) \right) f_3 \\
&= w_1 \left(\frac{Rn}{(Rn+Cn)} \right) \left(\sum_x \left[\sum_{i=1}^m \sigma_i \frac{q_{i,l_i}^l(q_l)}{f_k} \right]_{\min-\max} \times \frac{f_{x-\min-\max}(q_l)}{\sum_x f_{x-\min-\max}(q_l)} \right) \\
&\quad + \left(1 / \left(\frac{Rn}{Rn+Cn} + 1 \right) \right) \left(\sum_{i=1}^n \left(\sum_{j=1}^m t_0^{p,q} (1 + \alpha (\frac{x_{p,q}}{C_{p,q}})^\beta) + \sum_{k=1}^l t_c^k \right) \times V_{path,i} \right) \\
&\quad + \ln \left(\left(\frac{Rn}{Rn+Cn} + 1 \right) \right) (\phi T(t) + \alpha D_1 - C_1 + f + \alpha_2 - C_2 - kC)
\end{aligned} \tag{21}$$

5.3 问题 3 模型的建立与求解

5.3.1 基本分析

考虑各类型小区开放前后周边道路的影响，首先要构建开放程度不同的多类小区，通过 BPR 模型^[2]同时考虑道路结构，再计算小区开放后道路的阻抗函数，利用 MATLAB 软件计算不同小区模型、不同车流量、不同道路结构下小区开放后的车辆通行能力，比较不同情况下开放后的车辆通行能力，得到小区结构、周边道路结构、车流量对于小区开放效果的影响。

5.3.2 预测模型

(1) 灰色系统预测

灰色系统^[3]是指部分信息已知部分信息未知的系统，通过已知的数据得到预测未来的数据。 $q^{(0)}(t)$ 是收集的过去的车流量数据，通过累加生成 $q^{(1)}(k) = \sum_{i=1}^k q^{(0)}(i)$, $k=1,2,\dots,n$ ，得到新的生成数列 $q^{(1)}(t) = \{q^{(1)}(1), q^{(1)}(2), \dots, q^{(1)}(n)\}$ ，满足如下灰色预测的微分方程 $GM(1,1)$ ，白化形式为：

$$\frac{dq^{(1)}(t)}{dt} + aq^{(1)}(t) = b \tag{28}$$

(a, b 称为辨识参数)

$z^1(i)$ 为 $dq^{(1)}(t)/dt$ 在第 i 时刻的背景值， $z^1(i) = 1/2(q^{(1)}(i) + q^{(1)}(i-1))$, $i=2, \dots, n$

用最小二乘估计方法估计参数 a, b ，令 $B = \begin{pmatrix} q^{(1)}(2) & -1 \\ q^{(1)}(3) & -1 \\ \dots & \dots \\ q^{(1)}(n) & -1 \end{pmatrix}$, $Y = \begin{pmatrix} -q^{(0)}(2) \\ -q^{(0)}(3) \\ \dots \\ -q^{(0)}(n) \end{pmatrix}$,

$\alpha = (a, b)^T = (B^T B)^{-1} B^T Y$ ，估计出来的参数得到 $GM(1,1)$ 的方程为：

$$\hat{q}^{(1)}(t) = (q^{(0)}(1) - b/a) e^{-a(t-t_0)} + b/a \tag{29}$$

再做累减生成得到预测的序列：

$$\hat{q}_1^{(0)}(t) = (q^{(0)}(1) - b/a)(1 - e^{-a}) e^{-at}, t=1, 2, \dots, n, \dots \tag{30}$$

(2) 神经网络预测

BP 神经网络预测模型^[4]主要特点是：信号前向传递，而误差后向传播。如果输出层的

结果不是期望输出，则转入反向传播，根据预测误差调整网络权值和阈值，从而使 BP 神经网络预测输出逐渐接近期望输出。

step1. 构造样本 $\{P_1, t_1\}, \{P_2, t_2\}, \dots, \{P_n, t_n\}$ ，其中 P_i ， $i=1, 2, \dots, n$ 是过去的车流量数据， t_i ， $i=1, 2, \dots, n$ 是我们预先定的未来数据；对样本用最大最小方法进行归一化为 $[0, 1]$ 之间的数。

Step2. 我们构建神经网络，如下图：

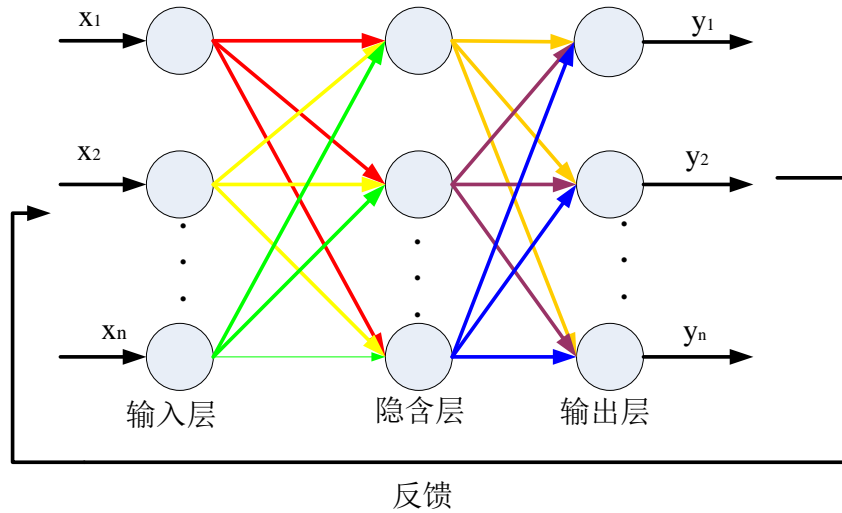


图 12 神经网络学习过程图

Step3. 通过 tansig 函数 $f(x) = \frac{2}{1+e^{-2x}} - 1$ 确定每一层神经元的传递函数，其中 x 是 BP 神经网络输入值。

Step4. 用 MATLAB 对其进行数据训练得到矩阵 net ，由此得预测结果 $q_2(t) = sim(net, x)$

(sim 是 MATLAB 中求 BP 神经网络预测结果的函数)。

Step5. 再对预测结果进行反归一化，得到最终的预测结果 $\hat{q}_2(t)$ 。

(3) ARIMA 模型预测

利用 spss 软件，通过数据转换，使数据成为零均值、无趋势的平稳序列。构建 $ARIMA(p, d, q)$ 模型。(p : 自回归过程阶数; d : 差分的阶数; q : 移动平均过程的阶数)

车流量的数据在各阶段是具有平稳性的，所以下一步是选择一个合理的模型来拟合要分析的时间序列，再用时间序列的数据，估计参数。最后建立模型，得到预测数据 $\hat{q}_3(t)$ 。

(4) 基于 IOWA 算子^[5-6]的组合预测模型

定义 $IOWA_w: R^n \times R^n \rightarrow R$ 函数， $\langle v_1, a_1 \rangle, \langle v_2, a_2 \rangle, \dots, \langle v_n, a_n \rangle$ 为 n 个二维数组，

$W=(w_1, w_2, \dots, w_n)^T$ 是与 $IOWA_W$ 有关的加权向量, $\sum_{i=1}^n w_i = 1, w_i \geq 0, i=1, 2, \dots, n$, 令

$$IOWA_W(\langle v_1, a_1 \rangle \langle v_2, a_2 \rangle \dots \langle v_n, a_n \rangle) = \sum_{i=1}^n w_i a_{index(i)} \quad (31)$$

其中 $a_{index(i)}$ 是 a_1, a_2, \dots, a_n 按从大到小的顺序排列的第 i 个大的数。

假设 $\{q_t\}_{t=1}^n$ 为真实原始数据, 现在有 3 种预测方法 \hat{q}_{it} , $i=1, 2, 3$, $t=1, 2, 3, \dots, n$, 表

$$\text{示第 } i \text{ 中预测方法、第 } t \text{ 时刻预测值, } e_{it} = \begin{cases} -1, (q_t - q_{it})/q_t < -1 \\ (q_t - q_{it})/q_t, -1 \leq (q_t - q_{it})/q_t \leq 1 \\ 1, (q_t - q_{it})/q_t > 1 \end{cases}, \text{ 显然}$$

$0 \leq |e_{it}| \leq 1$, $v_{it} = 1 - |e_{it}|$ 称为第 i 种预测方法在第 t 时刻的预测精度, 得到新的二维向量数

组 $(\langle v_{1t}, \hat{q}_{1t} \rangle \langle v_{2t}, \hat{q}_{2t} \rangle \dots \langle v_{nt}, \hat{q}_{nt} \rangle)$ 预测值。为了使误差平方和最小化, 设

$e_{index(it)} = q_t - q_{index(it)}$, 则

$$\begin{aligned} \min \sum_{t=1}^n (\hat{q}_t - q_t)^2 &= \sum_{t=1}^n \left(\sum_{i=1}^n w_i \hat{q}_{index(it)} - q_t \right)^2, \\ &= \sum_{i=1}^n \sum_{j=1}^n w_i w_j \left(\sum_{t=1}^n e_{index(it)} q_{index(jt)} \right) \\ &\quad s.t. \begin{cases} \sum_{i=1}^n w_i = 1 \\ w_i \geq 0 \end{cases}, \end{aligned} \quad (32)$$

由此得到权重值, 再利用 *Matlab* 程序得到最后预测值。

利用预测结果, 得到各类型小区不开放的假设下, 未来几年的车流量, 发现某些类型小区车流量明显增长或者道路饱和度仍然维持在较高水平, 而有些类型小区附近的车流量并无明显增长或是道路服务水平仍然处于一级优等状态。若预测结果的道路饱和度大于 1.0, 则该类小区在实行开放式政策后交通压力会明显减小, 应该实行开放, 开放效果最好; 若道路饱和度在 0.7 ~ 1.0 之间, 则开放小区后, 周边交通压力会有所减小, 车辆通行顺畅度加大, 可以缓解高峰时期道路拥挤的情况, 开放效果良好, 该类型小区可以实行开放政策; 若道路饱和度 < 0.7, 说明未来车流量对该类型小区未来周边道路的交通压力很小, 开放效果不明显, 开放政策不需要应用在该类型小区。

5.3.3 模型的建立与求解

一共测试 5 个模型, 小区正方形。模型 1 小区未开放, 模型 2-5 为不同道路结构的开放小区模型, 如图 3 所示:

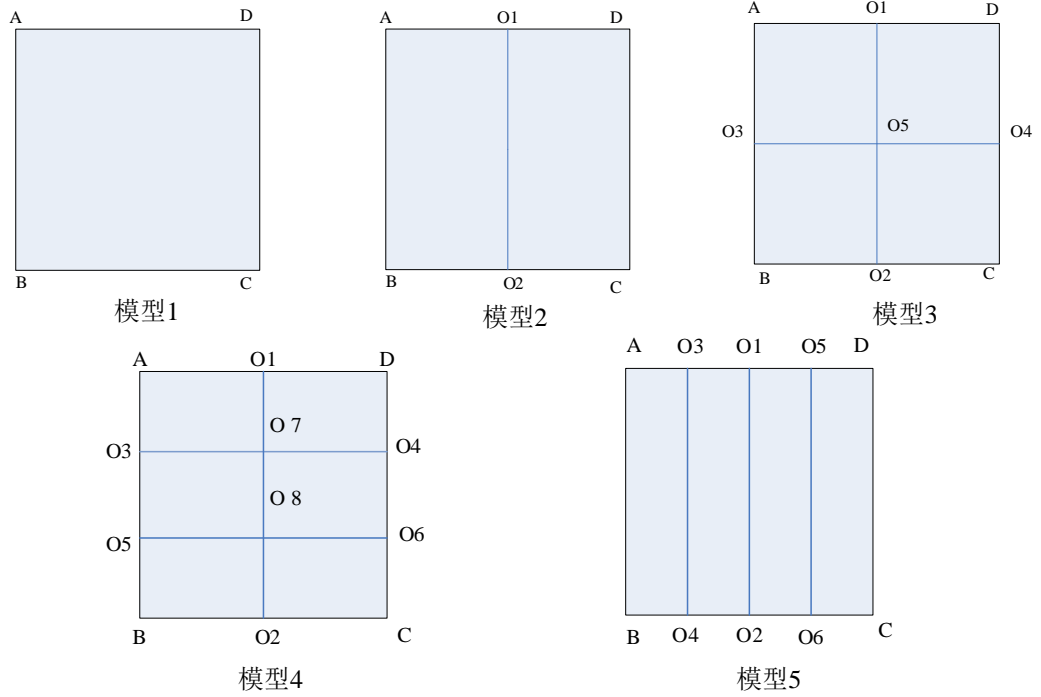


图3 5种小区模型（模型1作为未开放模型进行对比）

Step 1 路段阻抗BPR函数建立

首先假设孤立方型小区 $ABCD$ ，如图所示，小区边长 3 km ，四周边是主路，汽车平均时速 30 km/h ，则自由运行 3 km 时间 $t_0=360\text{ s}$ ，设计通行能力 $c=1300\text{ pcu}(\text{km}/\text{ln})$ ，小路段实际阻抗函数为：

$$t^{i,j} = t_0^{i,j} (1 + \alpha (\frac{x_{i,j}}{C_{i,j}})^\beta) \quad (22)$$

$t_0^{i,j}$ 是路段 I,J 上自由流时的走行时间, $C_{i,j}$ 是路段 I,J 的设计通行能力； $x_{i,j}$ 是路段 i,j 的实际车流量； α, β 是路段 I,J 的延误参数，采用实际考察数据回归分析得 $\alpha=0.18, \beta=4.01$ ；小区内道路是支路，汽车平均时速 15 km/h ，则自由运行 3 km 时间 $t_0=720\text{ s}$ ，设计通行能力 $c=800\text{ pcu}(\text{km}/\text{ln})$

小区的指路要考虑到自行车和行人的影响，对 $C_{i,j}$ 进行修正，即建立修正的 BPR 模型：

$$t^{i,j} = \begin{cases} t_0^{i,j} (1 + \alpha (\frac{x_{i,j}}{\eta \eta_1 C_{i,j}})^\beta), & 0 \leq \eta_1 \leq 1 \\ t_0^{i,j} (1 + \alpha (\frac{\eta_1 x_{i,j}}{\eta C_{i,j}})^\beta), & \eta_1 \geq 1 \end{cases} \quad (23)$$

其中 $t_0^{i,j}$ 是路段 I,J 上自由流时的走行时间； $C_{i,j}$ 是路段 I,J 的设计通行能力； $x_{i,j}$ 是路段 I,J 的实际车流量； α, β 是路段 I,J 的延误参数，取回归的结果， $\alpha=0.18, \beta=4.01$ ；

η 为行人的修正干扰系数，行人的干扰程度与系数选取的关系见表 1；

表 1 行人干扰修正系数建议值

干扰程度	很严重	严重	较严重	一般	很小	无
η	0.5	0.6	0.7	0.8	0.9	1.0

η_1 为自行车对机动车的干扰。其计算公式为：

$$\eta_1 = 0.8 - (q_{bi} / Q_{bi} + 0.5 - w_2) / w_1 \quad (24)$$

其中， q_{bi} 是实测道路上的自行车量（包括电动车）； Q_{bi} 是非机动车道路设计的每米通行能力； w_1 是单向非机动车宽度； w_2 是单向机动车宽度。

我们根据某试点开放小区的数据，代入公式计算得到合理的参数选择， $\eta = 0.8, \eta_1 = 0.8$ 。

Step 2 交叉口阻抗函数计算

我们针对不同的交叉路口类型，在路口处不同的运动状态，我们采用不同的计算参数和计算公式从而得到各个路口各种运动状态的阻抗函数 t 。

(1) 主路和支路的路口

此路口是丁字路口在，车辆在此处的运动状态存在有从主路右转进入支路，在主路直行，从支路左转进入主路三种情况，分别计算。

1) 右转，我们考虑右转专用道的阻抗：

$$t_{right} \approx 3 \sim 3.6s \text{ 取 } 3.4s$$

2) 左转，考虑直行和左转车道分离，计算左转的阻抗。

$$t_{left} = \frac{T_p}{n_{left}}, \quad (25)$$

其中， T_p 是一个信号灯周期，取 90s， n_{left} 是一个周期允许直行的车辆数。

$$n_{left} = \frac{t_l - \frac{v_{left}}{2 * a_{left}}}{t_{veh}}, \quad t_l \text{ 是一个周期内专用左转车的时间, 取 } 30s, v_{left} \text{ 左转车速度, 取 } 15km/h,$$

a_{left} 是左转平均加速度，取 $0.6m/s^2$ 。 t_{veh} 是直行车经过停车线的车头时距，取 5s

经过计算得， $t_{left} = 16.963s$

3) 直行，考虑直行的绿灯时间较长，直行和左转车道分离。

$$t_{stra} = \frac{T_p}{n_{stra}}, \quad (26)$$

其中， T_p 是一个信号灯周期，取 90s， n_{stra} 是一个周期允许左转的车辆数。

$$n_{stra} = \frac{t_s - \frac{v_{stra}}{2 * a_{stra}}}{t_{sveh}}, \quad t_s \text{ 是一个周期内专用直行车的时间, 取 } 45s, \quad v_{stra} \text{ 直行车速度, 取}$$

15km/h, a_{stra} 是左转平均加速度, 取 $0.6m/s^2$ 。 t_{sveh} 是左转车经过停车线的车头时距, 取 2.5s。经过计算得, $t_{stra} = 8.482s$

(2) 支路和支路的交叉口

此路口是十字路口, 考虑车辆在这其中的直行, 右转, 左转三种运动状态。小区内道路一般比较狭窄, 会存在直行和转向的车辆混行的状态。要对公式作修正, 考虑到左转车, 右转车占直行的比例, 再考虑到相互混行的时候的折损, 引入折减系数。具体公式为:

$$t'_{left} = t_{left} (1 - \frac{n_l}{2}) K, t'_{right} = t_{right} (1 - \frac{n_r}{2}) K, t'_{stra} = t_{stra} (1 - \frac{n_s}{2}) K, \quad (27)$$

代入实际参数计算得到:

$$t_{left} = 29.690s, t_{stra} = 14.835s, t_{right} = 6.300s$$

(3) 主路和主路的交叉路口

我们也要考虑车流转向的阻抗, 利用相似的计算方法得到主路和主路交叉口各种运动状态的阻抗函数。

Step 3 整张路网的车流量分布

给定小区结构周围路网消息, 我们首先深度优先递归搜索出所有可能路线, 并建立各个路线的阻抗函数。在自然状态下, 当路网加入车流量^[1], 其倾向于阻抗最小的路线, 最后整个路网的分布呈现一种近平衡状态。为了模拟这种真实的状态, 我们采用容量限制——增量加载的方法。其基本思路是将车流量分成小份, 每次都小份车流量加入阻抗最小的路线, 并更新路网的车流分布和路线阻抗, 直到加入所有的车流量。由此我们得到了整张路网的车流量分布。

Step 4 计算交通效率, 路网阻抗和效用函数, 评价路网通行能力

我们在得到路网的车流量分布之后, 可以计算出路网的交通效率 f_1 和道路阻抗 f_2 。我们再计算出效用函数 f_3 。根据加权求和关系式 (15) $f = w_1 f_1 + w_2 f_2 + w_3 f_3$, 我们就得到了评价通行能力的总指标。由此我们可以讨论通行能力和小区面积, 小区结构, 小区行人自行车密度, 小区内道路宽度, 小区周边道路结构, 周边道路通行能力, 车流量等众多因素对小区周围路网能力的影响。据此, 我们可以得到是否应该开放小区以及开放小区的适用情况。

Step 5 探讨不同因素对小区开放效果的影响

我们采用不同的参数, 来模拟计算五种模型的通行能力, 从而探讨不同因素对小区开放效果的影响。

a) 小区面积（宽度）和车流量共同影响。

我们将车流量从 3500pcu 向 5100pcu 变化，小区宽度从 0.5km 向 6km 变化；用 matlab 程序计算出五种模型在不同参数下的综合通行能力。为了更加清晰的表现小区开放的效果，我们计算模型 2-5 相对模型 1 的综合通行能力的变化百分比。由此，变化百分比正值表示提高，且值越大代表开放小区对综合交通能力提高的效果越明显。见图 4~7

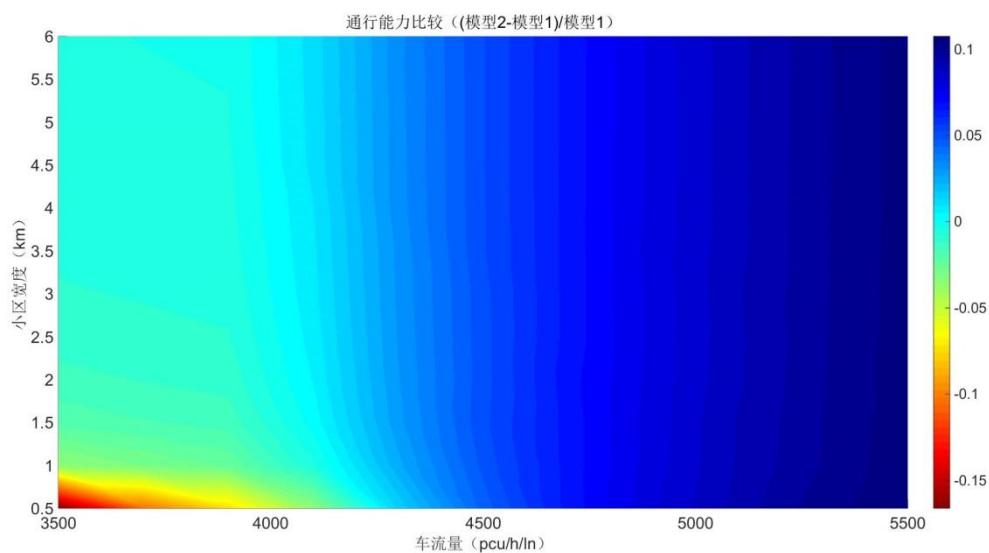


图 4 模型 2 与模型 1 的通行能力比较

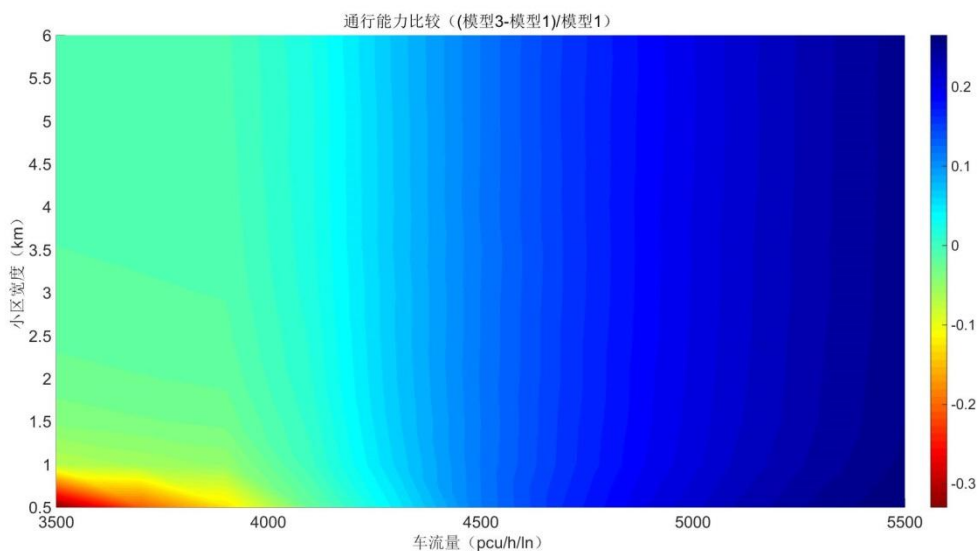


图 5 模型 3 与模型 1 的通行能力比较

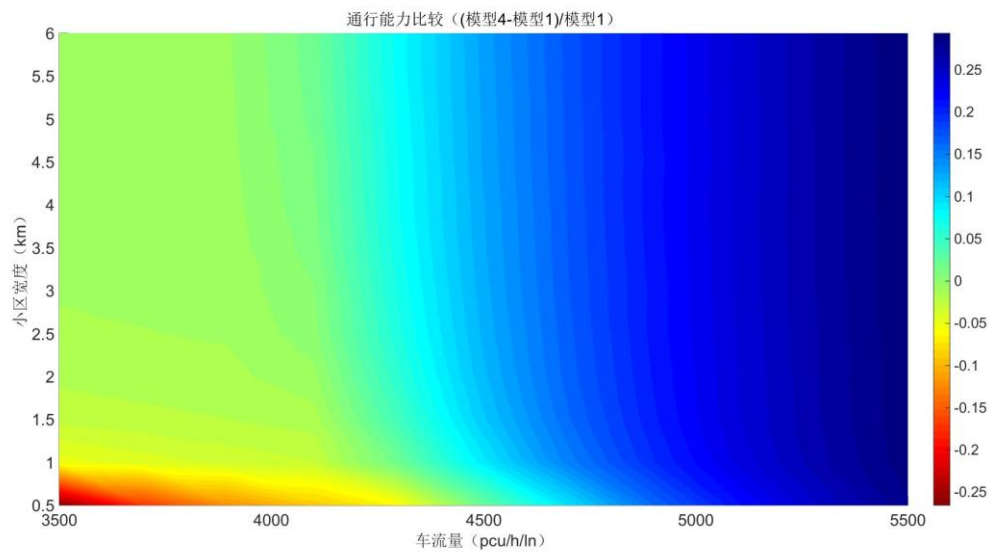


图 6 模型 4 与模型 1 的通行能力比较

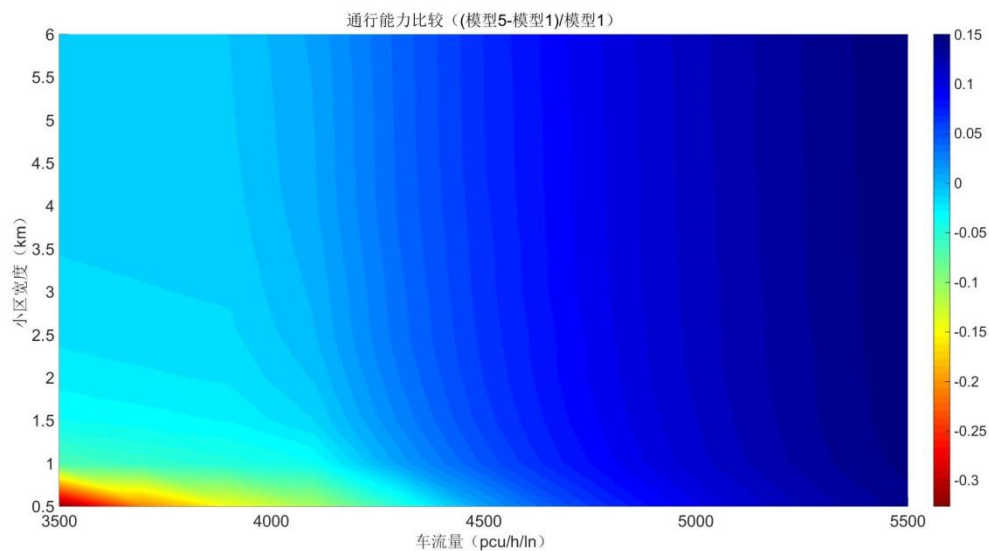


图 7 模型 5 与模型 1 的通行能力比较

由四张图片可以发现，车流量和小区面积对综合通行能力的提高有很大的影响，一般情况下，小区面积越大，车流量越大，开放小区的提高作用就越明显。在车流量比较小，小区面积表较小的情况下，开放小区甚至会造成降低小区周边的综合通行能力，这是符合常识的。对比四幅图，不同的小区结构对综合交通能力的提高幅度是不同的，模型 4 的效果最好，模型 2 的效果最差。这也是可以想象的。

b) 车流量的单因素影响

为了比较不同车流量下的不同模型的提高作用的差别，做出五种模型的综合通行能力随车流量的变化，见图 8，

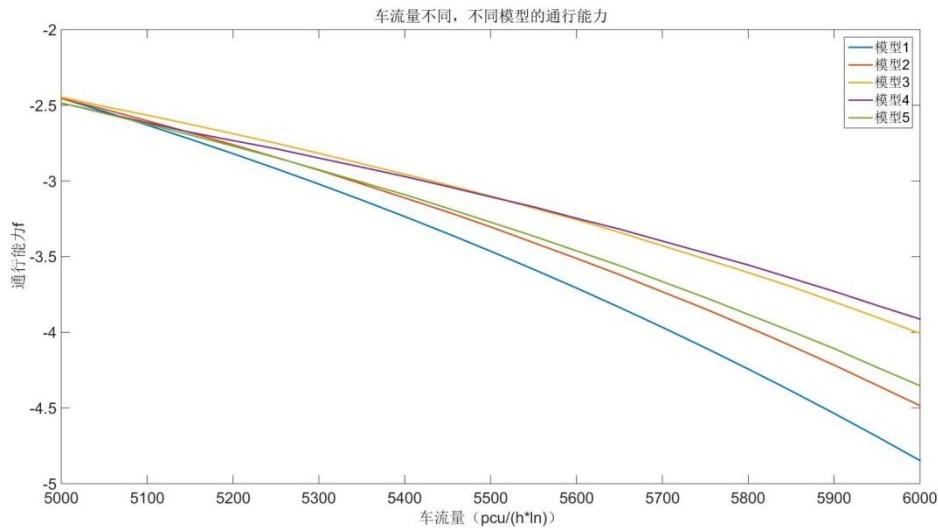


图 8 不同车流量下，模型 1-5 的综合通行能力

由比较图，可以看出，随着车流量的增大，综合通行能力在大大的下降。比较五种模型，模型的综合通行能力最低。说明模型 2-5，开放小区对综合通行能力均有提高作用。在车流量较低的时候，模型 3 的效果是最好的，但是当车流量增加到 $5400 \text{ pcu} / (h * \ln)$ 左右时，模型 4 的效果好于模型 3。

c) 小区面积的单因素影响

为了比较不同小区面积下，不同模型提高作用差别，做出五种模型的综合通行能力随小区面积的变化，见图 9。

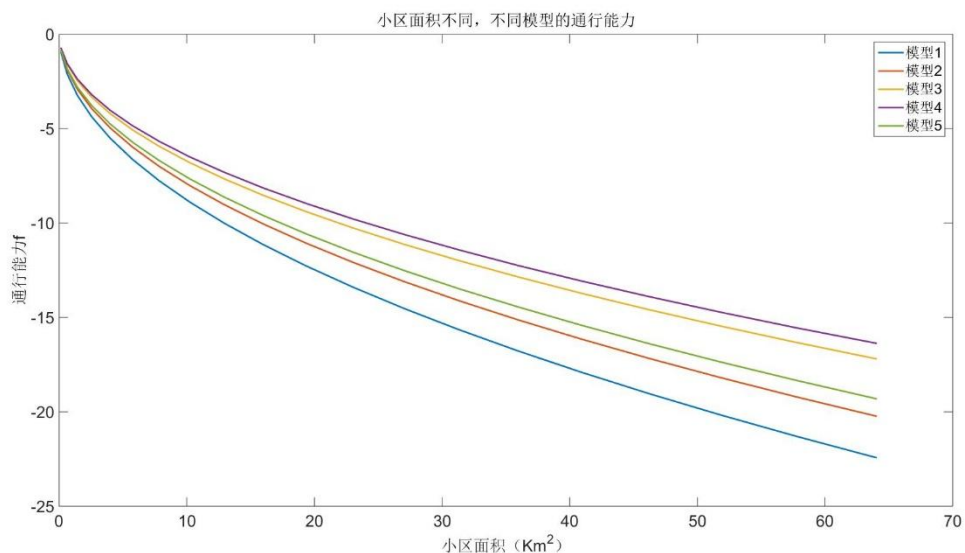


图 9 不同小区面积下，模型 1-5 的综合通行能力

小区面积越大，综合通行能力越差。模型 2-5，开放小区对综合交通能力具有提高作用且小区面积越大，提高作用越明显。模型 4 的效果最好。

d) 周边道路的影响

小区开放的效果不但有小区自身的结构面积等因素有关，还与周边道路的情况有关。为了验证周边道路对小区开放效果的影响。我们改变周边道路的设计通行能力，比较不同模型的综合通行能力。

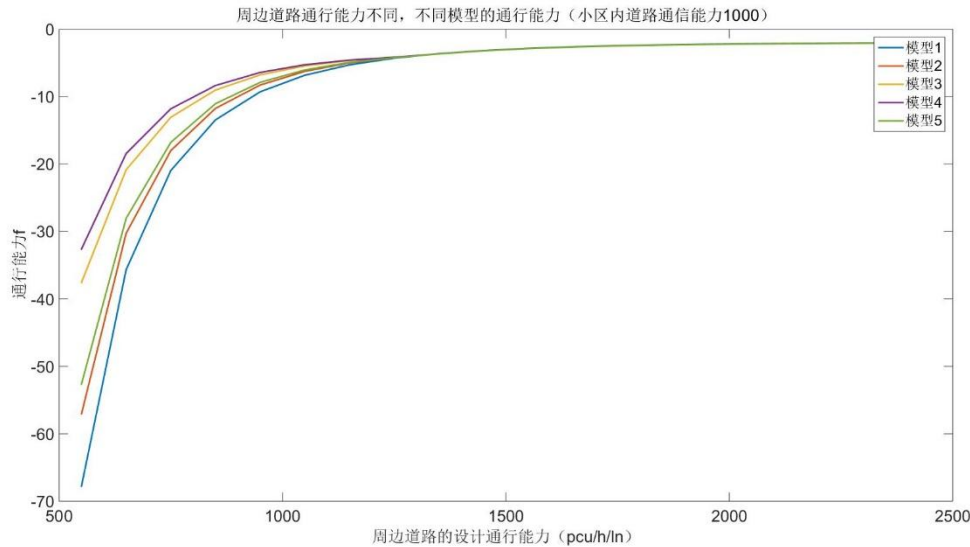


图 10 不同道路通行能力，模型 1-5 的综合通行能力

由图 10 可以看出，当周边道路的通行能力比较好的时候（大于 $1500\text{pcu}/(\text{h} \cdot \text{ln})$ ），变相当于车流量较小，开放小区的影响比较小。当周边道路的通行能力比较差的时候，模型 2-5 的开放效果明显，且周边道路通行能力越差，小区开放的效果越明显。四种开放模型中，模型 4 的效果最好

e) 车流量分布不均匀

考虑处于过境区域的小区，例如接近市郊的区域。往往在不同的时段，双向车流量并不处于一种均匀的状态。早高峰，进城方向车流量远远大于出城方向车流量，晚高峰则反之。这个时候我们在开放小区的时候，简单粗暴的把小区内道路平均分配显然不够合理。如果我们对小区内的支路采取分时段单行道，将更多的道路资源让给需求更大密集车流方向，将使得交通网的通行能力大大提高。为了验证我们的想法，我们使用模型 2 和模型 4 进行模拟。假设密集方向车流量是稀疏方向车流量的 10 倍，改变密集方向车流量，计算两种模型平均分配和分时段单行道的策略下的通行能力。见图 11

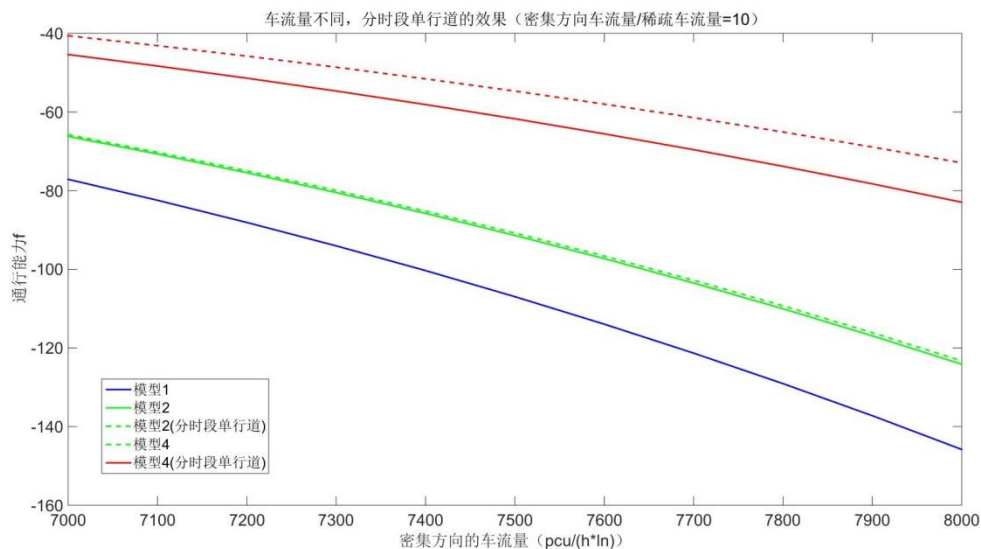


图 11 分时段单行道策略下，模型 2 和模型 4 的通行能力的提高

可以看出，分时段单行道对于提高开放小区周边的综合通行能力有积极的因素。模型 4 的结果看出，分时段单行道对于综合通行能力有很大的提高。另外，这种提高对于模型的结构也比较敏感。

f)其他因素：

在这些显式的影响因素之外。还有很多隐式的影响因素。例如，小区内的自行车行人密度通过影响小区内道路的行人和自行车修正系数，影响着我们的结果，可以预计的是，小区内自行车行人密度越小，开放小区对提高通行能力的效果越明显。在此之外，小区内道路的宽度越大，实际通行能力越大，小区内的结构分布越合理（小区内平均车速提高），小区开放的效果都越明显。

综上，我们可以得到结论，小区的面积越大，车流量越大，周边道路的通行能力越差，小区内的结构越合理，小区行人自行车密度越小，开放小区的效果越好。一般情况下，树形结构的小区（模型 4）开放效果最好。与此同时我们也要注意，开放小区不是一概而论的，是否开放小区，具体何种开放方案也要根据实际情况选定。

5.4 问题 4 模型的建立与求解

5.4.1 基本分析

交通通行的压力可由车流量来衡量，所以我们从交通通行的角度提出的意见要考虑到小区是否该开放，该如何开放具体会对周边未来的车流量产生的影响，因此首先需预测小区周边未来的车流量，预测方法采用 *TOPSIS* 组合预测，用它来对灰色预测、神经网络预测、ARIMA 预测三个单项预测结果进行综合。

5.4.2 开放方案的定量选择——基于 *TOPSIS* 决策方法

使用 *TOPSIS* 决策方法构造正理想解与负理想解，再比较各方案与正的贴近程度，对

各方案赋予综合评价价值，来选择最优决策方案。最优决策方案往往距离理想解最近，而距离负理想解最远。小区道路开放方式的综合效益利用 *TOPSIS* 多属性决策方法来判断哪种方案更加可用。

Step 1 构建方案集、属性集、决策矩阵

设方案集为 $S = \{S_1, S_2, \dots, S_m\}$ ；属性集为 $P = \{P_1, P_2, \dots, P_n\}$ ；多属性决策矩阵 $X = (x_{ij})_{m \times n}$ ，属性集包括效益型属性即属性值越大越好的指标和成本性属性即属性值越小越好的指标。多属性决策矩阵 $X = (x_{ij})_{m \times n}$ ，其中 x_{ij} 为第 i 个方案在第 j 个属性的指标值， $i = 1, 2, \dots, m$ ， $j = 1, 2, \dots, n$ 。

在本文中 S_i 表示改进小区的方案，属性集 P_1 表示路网交通效率； P_2 表示道路抗阻； P_3 表示综合效用。通过对第 i 个方案各项参数的计算得到三项属性的值。

Step 2 构造标准化的决策矩阵

设标准化的决策矩阵为 $Y = (y_{ij})_{m \times n}$ 。对于效益型属性，令 $y_{ij} = (x_{ij} - x_j^{\min}) / (x_j^{\max} - x_j^{\min})$ ；对于成本性属性， $y_{ij} = (x_j^{\max} - x_{ij}) / (x_j^{\max} - x_j^{\min})$ ， x_j^{\max} ， x_j^{\min} 分别为第 j 个属性 P_j 的最小值和最大值。

本文中 P_1 ， P_3 为效益型指标； P_2 为成本性指标。在这一步中需对这些指标按照相应方法进行标准化。

Step 3 确定指标的权重

根据前文二元语义得出的权重向量，构造加权标准化后的决策矩阵 $Z = (z_{ij})_{m \times n}$ ，其中 $z_{ij} = w_j y_{ij}$

Step 4 确定正理想解 A^+ 和负理想解 A^-

设 $A^+ = (z_1^+, z_2^+, \dots, z_n^+)$, $A^- = (z_1^-, z_2^-, \dots, z_n^-)$ ，其中

$$z_j^+ = \begin{cases} \max_{1 \leq i \leq m} z_{ij}, j \in j^+ \\ \min_{1 \leq i \leq m} z_{ij}, j \in j^- \end{cases}, \quad z_j^- = \begin{cases} \min_{1 \leq i \leq m} z_{ij}, j \in j^+ \\ \max_{1 \leq i \leq m} z_{ij}, j \in j^- \end{cases} \quad (33)$$

Step 5 用欧氏距离计算各方案到正理想解和负理想解的距离

方案 S_i 到正理想解 A^+ 的距离 T_i^+ ， $T_i^+ = \sqrt{\sum_{j=1}^n (z_{ij} - z_j^+)^2}$ $i = 1, 2, \dots, m$ ；

方案 S_i 到负理想解 A^- 的距离 T_i^- ， $T_i^- = \sqrt{\sum_{j=1}^n (z_{ij} - z_j^-)^2}$ $i = 1, 2, \dots, m$ 。

Step 6 计算各方案与理想解的相对贴近度

$$\text{相对贴近度 } C_i = \frac{T_i^-}{T_i^+ + T_i^-}, i = 1, 2, \dots, m \quad (34)$$

Step 7 选择最优方案

得到综合属性评价后，把 C_i 按照从大到小排序，最大 C_i 所对应的方案为城市规划和交通管理部门可选择的最优方案。

5.4.3 开放方案的定性选择

(1) 车流量大小与开放小区的决策。根据局域 IOWA 算子的组合预测模型所计算的结果，车流量较大的小区相对于车流量较小的小区而言，在相同的开放成本下，缓解交通压力的效果更好。因此相关部门在车流量较大的情况下应该选择开放。

(2) 高车流量前提下的小区构造与开放小区的决策。随着车流量不断增大，各个小区都倾向于开放以缓解交通压力，但是不同模型小区的开放效果也是不同的。在车流量较低时，有关部门应该基于模型 3 去规划小区的开放方式。作为在车流量增加到一定程度后，基于模型 4 来开放小区效果更好。

(3) 小区面积与开放小区的决策。小区面积越大，对道路的阻塞作用越明显，开放小区带来的效益更大。因此对于面积较大的，更应该选择开放小区。

(4) 大面积前提下的小区构造与开放小区的决策。随着小区面积不断增大，各个小区都倾向于开放以缓解交通压力，但是不同模型的小区开放效果同样呈现出不同。计算结果显示有关部门应该基于模型 4 进行小区的开发。

(5) 外部道路通行能力与开放小区的决策。随着外部道路通行能力较强的增强，交通压力较小，此时以开放小区的方式来缓解交通压力的必要性较低。

(6) 外部道路通行能力低前提下小区构造与开放小区决策。在外部道路通行能力较低的情况下，各小区都倾向于开放，但是开放的效果是不同的。有关部门应该基于模型 4 进行开放，此时效果最好。

(7) 车流量和是否分时段单行道的决策。随着车流量的增大，一些在市郊的小区会存在早晚方向相反的车流，此时有关部门应当对基于模型 2、4 构造的小区规定分时段单行道，以此提高周围路段的通行能力；而对其余三种类型构造的小区，不应该开放，因为开放与否对通行能力的影响几乎无差别。

六 模型的评价和改进

6.1 模型的评价

6.1.1 模型的优点：

1.对多样化的道路通行能力影响因子采用粗糙集的属性约简方法，得到的影响因子比较可靠且全面。

2.对评价体系的建立采用了多种模型分别从不同的侧面对综合指标定量的反映，并且对多种模型的评价结果进行多目标综合评价，得到的结果更加可靠。

- 3.根据每个题目的不同,对全文的宏观模型进行修正,使模型更加严谨和符合实际。
- 4.对小区未来的交通压力进行了组合预测,使我们的建议有了更强的说服力。

6.1.2 模型的缺点

- 1.文章只考虑了小区是方形的情况,虽然讨论开放不同内部道路的情况,但仍然具有片面性,有一定的误差。
- 2.文章讨论小区开放建议时,只考虑了未来交通压力和小区内外结构,虽有代表性,但没有考虑会更加细致化的因素,有一定的主观性误差。

6.2 模型的改进

- 1.在对问题二的各目标进行权重的确定时,只采用了二元语义一种模型,结果难免产生偏大的误差,因此可利用多种权重确定方法,以进一步减小误差。
- 2.文章只考虑了小区是方形的情况,所以可以考虑将小区的结构分的更加细致,使结果更加符合真实情况。

参考文献:

- [1]李梅.存在拥堵路段的道路网交通流分配模型及其应用研究[D].北京交通大学,2016
- [2]王素欣,王雷震,高利,崔小光,陈雪梅.BRP 路阻函数的改进研究[J].武汉理工大学学报,2009,33(3):446-449
- [3]曾波,刘思峰,方志耕,谢乃明.灰色组合预测模型及其应用[J].2009,17(5):151-155
- [4]易帆.神经网络预测研究[D].西南交通大学,2015
- [5]陈华友,,金飞飞周礼刚.基于最大——最小贴近度的 IOWA 算子最优组合预测模型[J].数学的实践与认识,2013,43(7):110-116
- [6]陈华友,刘春林.基于 IOWA 算子的组合预测方法[J].预测,2003,22(6):61-65
- [7]李向朋.城市交通拥堵对策—封闭型小区交通开放研究[D].长沙理工大学,2014.
- [8]贾晓敏.城市道路通行能力影响因素研究[D].长安大学,2009.

附录:

matlab 程序

1、路线阻抗计算函数

```
function main
```

```
%      CoreNum=4; %调用的处理器个数
%      if matlabpool('size')<=0 %之前没有打开
%          matlabpool('open','local',CoreNum);
%      else %之前已经打开
%          disp('matlab pool already started');
%      end
```

Weight=[0.330915386212215,-0.536718459302900*1e-6,0.132366154484886];% 交通效率和路网阻抗的权重系数，考虑到使量级接近，路网的通行能力是路网交通效率和路网阻抗的加权平均。

```
%      nnum=9;%节点个数
%      nmap=[];%节点分布位置关系
%      t0=[];%各个小路段的自由运行时间，0 代表不通
%      w=t0;
%      w(w~=0)=1;%有向无环图的边权矩阵，1 代表边存在，0 代表不存在。
%      t0(t0==0)=nan;
%      C=[];%各个小路段的实际通行能力，0 代表不通
%      alp=[];%各个小路段的 alpha，0 代表不通
%      bet=[];%各个小路段的 beta，0 代表不通
%      eta=[];%各个小路段的 eta 行人干扰修正系数，0 代表不通
%      eta1=[];%各个小路段的 eta1，自行车干扰修正系数，0 代表不通
%      lpara=reshape([t0,C,alp,bet,eta,eta1],nnum,nnum,6);
%      spara=[];%各个路口的左转，直行，右转时间，第一行是左转，第二行直行，第三行右转,不存在某种运动状态则为 0
%      spara(sparp==0)=nan;
%      totvol=1300;%从 A 到 C 总共的车流量;
%      partn=1000;%迭代计算的时候总共的车流量分成的份数
f3=0;
Lrange=6;
Vrange=7000:100:8000;
Cp=1200;
```

```

tt1=zeros(length(Lrange),length(Vrange),length(Cp));
tt2=zeros(length(Lrange),length(Vrange),length(Cp));
tt3=zeros(length(Lrange),length(Vrange),length(Cp));
tt4=zeros(length(Lrange),length(Vrange),length(Cp));
tt5=zeros(length(Lrange),length(Vrange),length(Cp));
per1=zeros(length(Lrange),length(Vrange),length(Cp));
per2=zeros(length(Lrange),length(Vrange),length(Cp));
per3=zeros(length(Lrange),length(Vrange),length(Cp));
per4=zeros(length(Lrange),length(Vrange),length(Cp));
per5=zeros(length(Lrange),length(Vrange),length(Cp));
partn=1000;
%model1
nnum1=4;
len1=[0,1,1,0;
      0,0,0,1;
      0,0,0,1;
      0,0,0,0];
t01=[0,120,120,0;
     0,0,0,120;
     0,0,0,120;
     0,0,0,0];
w1=t01;
w1(w1~=0)=1;
[ways1,pathnum1]=waysearch(w1,1,nnum1);
%model2
nnum2=6;
len2=[0,0.5,0,1,0,0;
      0,0,0.5,0,1,0;
      0,0,0,0,0,1;
      0,0,0,0,0.5,0;
      0,0,0,0,0,0.5;
      0,0,0,0,0,0];
t02=[0,60,0,120,0,0;
     0,0,60,0,240,0;

```

```

0,0,0,0,0,120;
0,0,0,0,60,0;
0,0,0,0,0,60
0,0,0,0,0,0];
w2=t02;
w2(w2~=0)=1;
w21=w2;
w21(2,5)=0;
[ways2,pathnum2]=waysearch(w2,1,nnum2);
[ways21,pathnum21]=waysearch(w21,1,nnum2);
ways20=ways2;
%model3
nnum3=9;
len3=[0,0.5,0,0.5,0,0,0,0,0;
0,0,0.5,0,0.5,0,0,0,0;
0,0,0,0,0,0.5,0,0,0;
0,0,0,0,0.5,0,0.5,0,0;
0,0,0,0,0,0.5,0,0.5,0;
0,0,0,0,0,0,0,0,0.5;
0,0,0,0,0,0,0,0.5,0;
0,0,0,0,0,0,0,0,0.5;
0,0,0,0,0,0,0,0,0];
t03=[0,60,0,60,0,0,0,0,0;
0,0,60,0,120,0,0,0,0;
0,0,0,0,0,60,0,0,0;
0,0,0,0,120,0,60,0,0;
0,0,0,0,0,120,0,120,0;
0,0,0,0,0,0,0,0,60;
0,0,0,0,0,0,0,60,0;
0,0,0,0,0,0,0,0,60;
0,0,0,0,0,0,0,0,0];
w3=t03;
w3(w3~=0)=1;
[ways3,pathnum3]=waysearch(w3,1,nnum3);

```

```

%model4
nnum4=12;
t04=[0,60,0,40,0,0,0,0,0,0,0,0;
      0,0,60,0,80,0,0,0,0,0,0,0;
      0,0,0,0,40,0,0,0,0,0,0,0;
      0,0,0,0,120,0,40,0,0,0,0,0;
      0,0,0,0,120,0,80,0,0,0,0,0;
      0,0,0,0,0,0,40,0,0,0,0,0;
      0,0,0,0,0,0,120,0,40,0,0,0;
      0,0,0,0,0,0,120,0,80,0,0,0;
      0,0,0,0,0,0,0,0,40,0,0,40;
      0,0,0,0,0,0,0,0,0,60,0,0;
      0,0,0,0,0,0,0,0,0,0,60,0;
      0,0,0,0,0,0,0,0,0,0,0,0];
w4=t04;
w4(w4~=0)=1;
[ways4,pathnum4]=waysearch(w4,1,nnum4);
ways40=ways4;
w41=t04;
w41(w41==80|w41==120)=0;
w41(w41~=0)=1;
[ways41,pathnum4]=waysearch(w41,1,nnum4);
%model5
nnum5=10;
t05=[0,30,0,0,0,120,0,0,0,0;
      0,0,30,0,0,0,240,0,0,0;
      0,0,0,30,0,0,0,240,0,0;
      0,0,0,0,30,0,0,0,240,0;
      0,0,0,0,0,0,0,0,120;
      0,0,0,0,0,0,30,0,0,0;
      0,0,0,0,0,0,0,30,0,0;
      0,0,0,0,0,0,0,0,30,0;
      0,0,0,0,0,0,0,0,0,30;
      0,0,0,0,0,0,0,0,0,0];

```

```

w5=t05;
w5(w5~=0)=1;
[ways5,pathnum5]=waysearch(w5,1,nnum5);
s=0;
for i=1:length(Lrange)
    for j=1:length(Vrange)
        for k=1:length(Cp);
            s=s+1;
            display(s);

            L=Lrange(i);
            totvol1=Vrange(j);
            totvol2=totvol1/10;
            C0=Cp(k);
            %% model 1
            nmap=[1,2;3,4];
            len=L*len1;
            t0=L*t01;
            t0(t0==0)=nan;
            C=[0,1300,1300,0;
                0,0,0,1300;
                0,0,0,1300;
                0,0,0,0];
            C(C==1300)=C0;
            alp=0.18*ones(4,4);
            bet=4.01*ones(4,4);
            eta=ones(4,4);
            eta1=ones(4,4);
            lpara=reshape([t0,C,alp,bet,eta,eta1],nnum1,nnum1,6);
            spara=[0,0,16.963,0;
                0,0,0,0;
                0,3.4,0,0];

```

```

[wayvol11,travol11,wtime11,tottime11]=totaltime(nmap,lpara,spara,ways1,totvol1,partn);

```

```

[wayvol12,travol12,wtime12,tottime12]=totaltime(nmap,lpara,spara,ways1,totvol2,partn);

delta1=traeff(ways1,C,travol11,len,wayvol11,alp,bet);
delta2=traeff(ways1,C,travol12,len,wayvol12,alp,bet);
per1(i,j,k)=Weight*[delta1;tottime11;f3]+Weight*[delta2;tottime12;f3];

%% model 20
nmap=[1,2,3;4,5,6];
len=L*len2;
t0=L*t02;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0;
    0,0,1300,0,1000,0;
    0,0,0,0,0,1300;
    0,0,0,0,1300,0
    0,0,0,0,0,1300;
    0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(6,6);
bet=4.01*ones(6,6);
eta=ones(6,6);eta(2,5)=0.8;
eta1=ones(6,6);eta1(2,5)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum2,nnum2,6);
spara=[0,0,0,16.963,16.963,0;
    0,8.482,0,0,8.482,0;
    0,3.4,3.4,0,0,0];

[wayvol201,travol201,wtime201,tottime201]=totaltime(nmap,lpara,spara,ways20,totvol1,partn);

[wayvol202,travol202,wtime202,tottime202]=totaltime(nmap,lpara,spara,ways20,totvol2,partn);
delta1=traeff(ways20,C,travol201,len,wayvol201,alp,bet);
delta2=traeff(ways20,C,travol202,len,wayvol202,alp,bet);
per2(i,j,k)=Weight*[delta1;tottime201;f3]+Weight*[delta2;tottime202;f3];

```



```

%% model 21
nmap=[1,2,3;4,5,6];
len=L*len2;
t0=L*t02;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0;
    0,0,1300,0,1000,0;
    0,0,0,0,0,1300;
    0,0,0,0,1300,0
    0,0,0,0,0,1300;
    0,0,0,0,0,0];
C(C==1300)=C0;
C2=C;
C1=C;C1(C1==1000)=2000;
alp=0.18*ones(6,6);
bet=4.01*ones(6,6);
eta=ones(6,6);eta(2,5)=0.8;
eta1=ones(6,6);eta1(2,5)=0.8;
lpara1=reshape([t0,C1,alp,bet,eta,eta1],nnum2,nnum2,6);
lpara2=reshape([t0,C2,alp,bet,eta,eta1],nnum2,nnum2,6);
spara=[0,0,0,16.963,16.963,0;
    0,8.482,0,0,8.482,0;
    0,3.4,3.4,0,0,0];

[wayvol211,travol211,wtime211,tottime211]=totaltime(nmap,lpara1,spara,ways20,totvol1,partn);

[wayvol212,travol212,wtime212,tottime212]=totaltime(nmap,lpara2,spara,ways21,totvol2,partn);
delta1=traeff(ways20,C1,travol211,len,wayvol211,alp,bet);
delta2=traeff(ways21,C2,travol212,len,wayvol212,alp,bet);
per3(i,j,k)=Weight*[delta1;tottime211;f3]+Weight*[delta2;tottime212;f3];

%% model 40
nmap=[1,2,3;4,5,6;7,8,9;10,11,12];
t0=L*t04;

```

```

len=t0/L;
len(len==60|len==120)=1/2;
len(len==40|len==80)=1/3;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0,0,0,0,0,0,0;
    0,0,1300,0,1000,0,0,0,0,0,0,0;
    0,0,0,0,0,1300,0,0,0,0,0,0;
    0,0,0,0,1000,0,1300,0,0,0,0,0;
    0,0,0,0,0,1000,0,1000,0,0,0,0;
    0,0,0,0,0,0,0,0,1300,0,0,0;
    0,0,0,0,0,0,0,1000,0,1300,0,0;
    0,0,0,0,0,0,0,0,1000,0,1000,0;
    0,0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(nnum4,nnum4);
bet=4.01*ones(nnum4,nnum4);
eta=ones(nnum4,nnum4);eta(C==1000)=0.8;
eta1=ones(nnum4,nnum4);eta1(C==1000)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum4,nnum4,6);
spara=[0,0,0,16.963,29.690,0,16.963,29.690,0,0,16.963,0;
        0,8.482,0,8.482,14.935,8.482,8.482,14.935,8.482,0,0,0;
        0,3.4,0,0,6.300,0,0,6.300,0,0,0,0];

```

```

[wayvol401,travol401,wtime201,tottime401]=totaltime(nmap,lpara,spara,ways40,totvol1,partn);

```

```

[wayvol402,travol402,wtime402,tottime402]=totaltime(nmap,lpara,spara,ways40,totvol2,partn);

```

```

delta1=traeff(ways40,C,travol401,len,wayvol401,alp,bet);
delta2=traeff(ways40,C,travol402,len,wayvol402,alp,bet);
per4(i,j,k)=Weight*[delta1;tottime401;f3]+Weight*[delta2;tottime402;f3];

```

```

%% model 41

```

```

nmap=[1,2,3;4,5,6;7,8,9;10,11,12];
t0=L*t04;
len=t0/L;
len(len==60|len==120)=1/2;
len(len==40|len==80)=1/3;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0,0,0,0,0,0;
    0,0,1300,0,1000,0,0,0,0,0,0;
    0,0,0,0,0,1300,0,0,0,0,0;
    0,0,0,0,1000,0,1300,0,0,0,0;
    0,0,0,0,0,1000,0,1000,0,0,0;
    0,0,0,0,0,0,0,0,1300,0,0;
    0,0,0,0,0,0,0,1000,0,1300,0;
    0,0,0,0,0,0,0,0,1000,0,1000;
    0,0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0,0,0];
C(C==1300)=C0;
C2=C;
C1=C;C1(C1==1000)=2000;
alp=0.18*ones(nnum4,nnum4);
bet=4.01*ones(nnum4,nnum4);
eta=ones(nnum4,nnum4);eta(C==1000)=0.8;
eta1=ones(nnum4,nnum4);eta1(C==1000)=0.8;
lpara1=reshape([t0,C1,alp,bet,eta,eta1],nnum4,nnum4,6);
lpara2=reshape([t0,C2,alp,bet,eta,eta1],nnum4,nnum4,6);
spara=[0,0,0,16.963,29.690,0,16.963,29.690,0,0,16.963,0;
        0,8.482,0,8.482,14.935,8.482,8.482,14.935,8.482,0,0,0;
        0,3.4,0,0,6.300,0,0,6.300,0,0,0,0];

```

```

[wayvol411,travol411,wtime411,tottime411]=totaltime(nmap,lpara1,spara,ways40,totvol1,partn);

```

```

[wayvol412,travol412,wtime412,tottime412]=totaltime(nmap,lpara2,spara,ways41,totvol2,partn);

```

```

        delta1=traeff(ways40,C1,travol411,len,wayvol411,alp,bet);
        delta2=traeff(ways41,C2,travol412,len,wayvol412,alp,bet);
        per5(i,j,k)=Weight*[delta1;totttime411;f3]+Weight*[delta2;totttime412;f3];

    end

end

end

per=[per1(:)';per2(:)';per3(:)';per4(:)';per5(:)'];
xlab='密集方向的车流量 (pcu/(h*ln)) ';
legstr={'模型 1';'模型 2';'模型 2(分时段单行道)';'模型 4';'模型 4(分时段单行道)'};
tit='车流量不同, 分时段单行道的效果 (密集车流量/稀疏车流量=10) ';
plotline(per,Vrange,xlab,legstr,tit)

for i=1:length(Lrange)-1
    for j=1:length(Vrange)
        for k=1:length(Cp);
            s=s+1;
            display(s);

            L=Lrange(i);
            totvol=Vrange(j);
            C0=Cp(k);
            %% model 1
            nmap=[1,2;3,4];
            len=L*len1;
            t0=L*t01;
            t0(t0==0)=nan;
            C=[0,1300,1300,0;
                0,0,0,1300;
                0,0,0,1300;
                0,0,0,0];
            C(C==1300)=C0;
            alp=0.18*ones(4,4);

```

```

bet=4.01*ones(4,4);
eta=ones(4,4);
eta1=ones(4,4);
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum1,nnum1,6);
spara=[0,0,16.963,0;
        0,0,0,0;
        0,3.4,0,0];

[wayvol1,travol1,wtime1,tottime1]=totaltime(nmap,lpara,spara,ways1,totvol,partn);

delta=traeff(ways1,C,travol1,len,wayvol1,alp,bet);
per1(i,j,k)=Weight*[delta;tottime1;f3];

%% model 2
nmap=[1,2,3;4,5,6];
len=L*len2;
t0=L*t02;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0;
    0,0,1300,0,1000,0;
    0,0,0,0,0,1300;
    0,0,0,0,1300,0;
    0,0,0,0,0,1300;
    0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(6,6);
bet=4.01*ones(6,6);
eta=ones(6,6);eta(2,5)=0.8;
eta1=ones(6,6);eta1(2,5)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum2,nnum2,6);
spara=[0,0,0,16.963,16.963,0;
        0,8.482,0,0,8.482,0;
        0,3.4,3.4,0,0,0];

[wayvol2,travol2,wtime2,tottime2]=totaltime(nmap,lpara,spara,ways2,totvol,partn);

```

```

delta=traeff(ways2,C,travol2,len,wayvol2,alp,bet);
per2(i,j,k)=Weight*[delta;tottime2;f3];

%% model 3
nmap=[1,2,3;4,5,6;7,8,9];
len=L*len3;
t0=L*t03;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0,0,0,0;
    0,0,1300,0,1000,0,0,0,0;
    0,0,0,0,1300,0,0,0,0;
    0,0,0,0,1000,0,1300,0,0;
    0,0,0,0,1000,0,1000,0,0;
    0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,1300,0,0;
    0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(9,9);
bet=4.01*ones(9,9);
eta=ones(9,9);eta(2,5)=0.8;eta(4,5)=0.8;eta(5,6)=0.8;eta(5,8)=0.8;
eta1=ones(9,9);eta1(2,5)=0.8;eta1(4,5)=0.8;eta1(5,6)=0.8;eta1(5,8)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum3,nnum3,6);
spara=[0,0,0,16.963,29.690,0,16.693,0,0;
    0,8.482,0,8.482,14.935,8.482,0,8.482,0;
    0,3.4,3.4,0,6.300,3.4,0,3.4,0];

[wayvol3,travol3,wtime3,tottime3]=totaltime(nmap,lpara,spara,ways3,totvol,partn);
delta=traeff(ways3,C,travol3,len,wayvol3,alp,bet);
per3(i,j,k)=Weight*[delta;tottime3;f3];

```

```

%% model 4

nmap=[1,2,3;4,5,6;7,8,9;10,11,12];
t0=L*t04;
len=t0/L;
len(len==60|len==120)=1/2;
len(len==40|len==80)=1/3;
t0(t0==0)=nan;
C=[0,1300,0,1300,0,0,0,0,0,0,0,0;
    0,0,1300,0,1000,0,0,0,0,0,0,0;
    0,0,0,0,1300,0,0,0,0,0,0,0;
    0,0,0,0,1000,0,1300,0,0,0,0,0;
    0,0,0,0,1000,0,1000,0,0,0,0,0;
    0,0,0,0,0,0,1300,0,0,0,0,0;
    0,0,0,0,0,0,1000,0,1300,0,0;
    0,0,0,0,0,0,0,1000,0,1000,0;
    0,0,0,0,0,0,0,0,1300,0,0;
    0,0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(nnum4,nnum4);
bet=4.01*ones(nnum4,nnum4);
eta=ones(nnum4,nnum4);eta(C==1000)=0.8;
eta1=ones(nnum4,nnum4);eta1(C==1000)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum4,nnum4,6);
spara=[0,0,0,16.963,29.690,0,16.963,29.690,0,0,16.963,0;
    0,8.482,0,8.482,14.935,8.482,8.482,14.935,8.482,0,0,0;
    0,3.4,0,0,6.300,0,0,6.300,0,0,0,0];

[wayvol4,travol4,wtime4,tottime4]=totaltime(nmap,lpara,spara,ways4,totvol,partn);
delta=traeff(ways4,C,travol4,len,wayvol4,alp,bet);
per4(i,j,k)=Weight*[delta;tottime4;f3];

```

```

%% model 5

```

```

nmap=[1,2,3,4,5;6,7,8,9,10];
t0=L*t05;
len=t0/L;
len(len==120|len==240)=1;
len(len==30)=1/4;
t0(t0==0)=nan;
C=[0,1300,0,0,0,1300,0,0,0,0;
    0,0,1300,0,0,0,1000,0,0,0;
    0,0,0,1300,0,0,0,1000,0,0;
    0,0,0,0,1300,0,0,0,1000,0;
    0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,1300,0,0,0;
    0,0,0,0,0,0,0,1300,0,0;
    0,0,0,0,0,0,0,0,1300,0;
    0,0,0,0,0,0,0,0,0,1300;
    0,0,0,0,0,0,0,0,0,0];
C(C==1300)=C0;
alp=0.18*ones(nnum5,nnum5);
bet=4.01*ones(nnum5,nnum5);
eta=ones(nnum5,nnum5);eta(C==1000)=0.8;
eta1=ones(nnum5,nnum5);eta1(C==1000)=0.8;
lpara=reshape([t0,C,alp,bet,eta,eta1],nnum5,nnum5,6);
spara=[0,0,0,0,0,0,16.963,16.963,16.963,0;
        0,8.482,8.482,8.482,0,0,8.482,8.482,8.482,0;
        0,3.4,3.4,3.4,0,0,0,0,0,0];

```

```

[wayvol5,travol5,wtime5,tottime5]=totaltime(nmap,lpara,spara,ways5,totvol,partn);
delta=traeff(ways5,C,travol5,len,wayvol5,alp,bet);
per5(i,j,k)=Weight*[delta;tottime5;f3];
tt1(i,j,k)=tottime1;
tt2(i,j,k)=tottime2;
tt3(i,j,k)=tottime3;
tt4(i,j,k)=tottime4;
tt5(i,j,k)=tottime5;

```



```

        end
    end
end
for i=2:5
    titstr=['通行能力比较 ((模型' num2str(i) '-模型 1)/模型 1) '];
    %plotimage(per1,eval(['per' num2str(i)]),Vrange,Lrange,titstr)
end
per=[per1(:);per2(:);per3(:);per4(:);per5(:)];
xlab='车流量 (pcu/(h*ln)) ';
legstr=['模型 1';'模型 2';'模型 3';'模型 4';'模型 5'];
tit='车流量不同，不同模型的通行能力';
plotline(per,Lrange,xlab,legstr,tit)
end
function plotline(per,xrange,xlab,leg,titstr)
    [pern,~]=size(per);
    figure
    for i=1:pern
        plot(xrange,per(i,:), 'linewidth',2);hold on;
    end
    title(titstr,'fontsize',20);
    set(gca,'fontsize',20);
    xlabel(xlab,'fontsize',20);
    ylabel('通行能力 f','fontsize',20);
    legend(leg);
end
function plotimage(per1,per2,Vrange,Lrange,titstr)
    pdiff=(per2-per1)./abs(per1);
    figure
    surf(Vrange,Lrange,pdiff);
    shading interp;
    view(2);
    axis tight;
    colormap(fliplr(jet));
    colorbar;

```

```

title(titstr,'fontsize',20);
set(gca,'fontsize',20);
xlabel('车流量 (pcu/h/ln) ','fontsize',20);
ylabel('小区宽度 (km) ','fontsize',20);
end
function delta=traeff(ways,C,trav,len,wayvol,alpha,beta)
% 根据公式计算路网的交通效率
[wayn,~]=size(ways);
fk=zeros(wayn,1);
sigmk=zeros(wayn,1);
for i=1:wayn
    if wayvol(i)==0
        continue;
    end
    way=ways(i,:);
    way=way(way>0);
    pn=length(way);
    sigm_a=zeros(pn-1,1);
    fkj=zeros(pn-1,1);
    for j=1:pn-1
        op=way(j);
        dp=way(j+1);
        q=wayvol(i);
        qa=trav(op,dp);
        Ca=C(op,dp);
        a=alpha(op,dp);
        b=beta(op,dp);
        l=len(op,dp);
        sigm_a(j)=qa/Ca/(1+a*(qa/Ca)^b);
        fkj(j)=q*l;
    end
    fk=sum(fkj);
    sigmk=sum(sigm_a.*fkj/fk);
end
end

```

```

    delta=sum(fk/sum(fk).*sigmk);
end

function [wayvol,travol,wtime,tottime]=totaltime(nmap,lpara,spara,ways,totvol,partn)

%给定各个小路段的参数，路线及总车流量，计算各个路线的车流量，各个路线的行程
时间，各个小路段的车流量分布，以及需要总时间

%参数

%N*N 的矩阵 行号是起点 列号是终点
%nmap:各个节点的分布位置关系，例如 N=6,nmap 可以表示为[1,2,3;4,5,6];
%w: 各段路的车流量 (N*N);
%lpara:路段参数 t0, C, alpha, beta,eta,eta1 (N*N*6);
%spara: 路口参数，左转 直行 右转 (N*3);
%ways: 一行是一个路线 (wayn 个路线，一行 N 个点，后边儿补零);
%totvol: 从 A 到 C 总共的车流量;
%partn: 迭代计算的时候总共的车流量分成的份数
%返回值

%wayvol 各个路线上的车流量
%travol 各个小路段的车流量
%wtime 各个路线上的行程时间
%tottime 总时间=车流量*对应行程时间

    partv=totvol/partn;
    pointn=length(nmap(:));
    [wayn,~]=size(ways);
    travol=zeros(pointn,pointn);
    wayvol=zeros(wayn,1);
    for i=1:partn
        time=waytime(nmap,travol,lpara,spara,ways);
        [~,ind]=min(time);
        way=ways(ind,:);
        way=way(way>0);
        waypn=length(way);
        wayvol(ind)=wayvol(ind)+partv;
        ini=way(1:end-1);
        inj=way(2:end);
        travol((inj-1)*pointn+ini)=travol((inj-1)*pointn+ini)+partv*ones(1,waypn-1);
    end

```

```

end

wtime=waytime(nmap,travol,lpara,spara,ways);

tottime=sum(wtime.*wayvol);

end

function [ways,pathnum]=waysearch(w,op,dp)
%给定有向图的边权矩阵及起止点，计算起止点之间的所有可能路线。
%w:有向图的边权矩阵，1 代表边存在，0 代表不存在。
%op:起点
%dp: 终点
%ways: 路线 一行是一个路线（wayn 个路线，一行 N 个点，后边儿补零）；
%pathnum: 路线条数

[np,~]=size(w);
next=find(w(op,:)==1);
nextn=length(next);
if op==dp
    pathnum=1;
    ways=zeros(1,np);
    ways(1:1)=dp;
    return;
else
    wayt=cell(nextn,1);
    patht=zeros(nextn,1);
    for i=1:nextn
        [wayt{i},patht(i)]=waysearch(w,next(i),dp);
    end
    pathnum=sum(patht);
    ways=zeros(pathnum,np);
    k=0;
    for i=1:nextn
        waytemp=wayt{i};
        for j=1:patht(i)
            k=k+1;
            ways(k,:)=[op,waytemp(j,1:end-1)];
        end
    end
end

```

end

end

2、使用 MSA 计算道路通信能力并比较

function time=waytime(nmap,travol,lpara,spara,way)

%给定各个小路段的参数，各个路段的车流量，以及各个路线，计算各个路线的行程时间。

%nmap:各个节点的分布位置关系，例如[1,2,3;4,5,6];

%w: 各段路的车流量 (N*N);

%lpara:路段参数 t0, C, alpha, beta,eta,etal (N*N*4);

%spara: 路口参数，左转 直行 右转 (N*3);

%way: 一行是一个线路

[wayn,~]=size(way);

time=zeros(wayn,1);

for i=1:wayn

ways=way(i,:);

ways=ways(ways>0);

ln=length(ways)-1;%每种路线的路段数目

sn=ln-1;%每种路线的实际交叉路口数目 除去起点和终点

lt=zeros(ln,1);%花费在每个路段间的时间

st=zeros(sn,1);%花费在每个交叉口的时间

for j=1:ln

op=ways(j);

dp=ways(j+1);

w=travol(op,dp);%路段车流量

t0=lpara(op,dp,1);%t0

C=lpara(op,dp,2);%C

a=lpara(op,dp,3);%alpha

b=lpara(op,dp,4);%beta

h=lpara(op,dp,5);%eta

h1=lpara(op,dp,6);%etal

lt(j)=ltime(w,t0,h,h1,C,a,b);

end

for k=1:sn

npre=ways(k);

```

        nnow=ways(k+1);
        nnext=ways(k+2);
        in=sparaind(nmap,npre,nnow,nnext);
        st(k)=spara(in,nnow);
    end
    time(i)=sum(lt)+sum(st);
end
end
function t=ltime(w,t0,h,h1,C,a,b)
% 计算各个小路段的形成时间
    if h1<=1
        t=t0*(1+a*(w/(h*h1*C))^b);
    else
        t=t0*(1+a*(h1*w/(h*C))^b);
    end
end
end
function in=sparaind(nmap,npre,nnow,nnext)
% 判断在某一路口的运行状态，1 左转，2，直行，3，右转
    [ind1i,ind1j]=find(nmap==npre);
    [ind2i,ind2j]=find(nmap==nnow);
    [ind3i,ind3j]=find(nmap==nnext);
    p=(ind3i-ind1i)*(ind3j-ind1j)*(ind3i-ind2i+ind2j-ind3j);
    in=p+2;
end
end

```

3.灰色预测MATLAB代码

```

clear all
close all
clc
x0=[3647 4837 4176 2318 4898 4484 4758 4642 3467];
m=3;
n=length(x0);
x1=zeros(1,n);
x1(1)=x0(1);
for i=2:n

```

```

        x1(i)=x1(i-1)+x0(i);
    end
    i=2:n;
        y(i-1)=x0(i);
    y=y';
    c=ones(1,n-1);
    for i=1:n-1;
        c(i)=-0.5*(x1(i)+x1(i+1));
    end
    B=[c' ones(n-1,1)];
    au=(B'*B)\B'*y;
    ago=ones(1,n+1+m);
    for i=1:n+1+m;
        ago(i)=(x0(1)-au(2)/au(1))*exp(-au(1)*(i-1))+au(2)/au(1);
    end
    yc(1)=ago(1);

    for i=1:n-1;
        yc(i+1)=ago(i+1)-ago(i);
    end
    error=ones(1,n-1);
    for i=2:n;
        error(i)=yc(i)-x0(i);
    end
    yc(1)=ago(1);
    yc=ones(1,n-1-m);
    for i=1:n-1+m;
        yc(i+1)=ago(i+1)-ago(i);
    end
    c=std(error)/std(x0);
    p=0;
    for i=2:n
        if abs(error(i)-mean(error))<0.6745*std(x0)
            p=p+1;

```

```

        end
    end
    p=p/(n-1);
    w1=min(abs(error));
    w2=max(abs(error));
    w=ones(1,n);
    for i=1:n;
        w(i)=(w1+0.5*w2)./(abs(error(i))+0.5*w2);
    end
    w=sum(w)/(n-1);

```

4.组合预测权重确定MATLAB代码

```

clear all
close all
clc
P=[0.5 0.25 0.75
0.8 0.5 1
0.2 0.05 0.6];
p1=P(1,:)/sum(P,1)*1/3;
p2=P(2,:)/sum(P,1)*1/3;
p3=P(3,:)/sum(P,1)*1/3;
r1=p1/(p1+p2+p3)
r2=p2/(p1+p2+p3)
r3=p3/(p1+p2+p3)

```