

城市相邻两交叉口信号配时优化

摘要

随着我国城市化速度的加快以及城市规模的不断扩大,交通供需矛盾日益突出,在城市交通网络中产生的交通拥堵现象日趋严重,合理的设计智能交通系统,优化相邻交叉口的信号配时问题将在很大程度上提高平面交叉口的通行能力和服务水平,减少城市交通网的交通延误,改善城市交通现状。

针对优化城市相邻两交叉口信号配时问题,我们首先从单点平面交叉口着手,建立了基于多目标优化的交叉口交通灯动态配时模型,用通行能力、延误、停车次数这三个指标通过加权作为多目标联合优化的参数指标,把排队长度作为模糊控制指标,并且追加有效绿灯时间,进而得到信号配时的联合优化函数,最终我们确定了在单点交叉口下的最佳周期 C 和各相位的绿灯有效时间。

其次,我们在单点的交叉口的基础上进一步分析了相邻两交叉口的情况,本文以简单绿波系统为例,建立一个基于基本交通流模型以及交叉路口的排队分析理论,以两个交叉路口的车辆总延误最小为目标,以主干道的绿波效应以及中间路段的车辆排队长始终小于其最大车辆排队长度为约束条件的非线性函数模型,并针对不同的交叉口饱和率进行信号配时优化,通过对路段和干线机动车流进行协调控制设计,在对路段和干线机动车流进行协调控制设计后,有效地减少了相交道路车流对干线车流的干扰,实现了邻近交叉路口信号灯的实时控制,保证了相交道路车流行车需求的同时,使干线车流获得最大的通行权。

再次,期望两相邻的交叉路口能更好的使协调配合工作,我们又相继建立了车辆延误模型来考察车均延误时间,采用了数解法以最大绿波带为优化目标通过寻找使得系统中各实际信号距理想信号的最大挪移量最小来对相位差进行优化设计,最终借助题中表一数据进行了仿真检验,实现了在保证交通流量的情况下改善了车均延误时间,最后我们推广到了多个相关交叉口并借助一例进行了应用,使模型更具普遍性。

之后,我们做了从时间复杂度、空间复杂度、收敛性三个方面总结性的对比分析工作,集中展示了在应用本文模型改善后的交通信号配时方案,并进行了简单的仿真检验,可以做为指导提高实际平面交叉口的通行能力和服务水平,减少城市交通网的交通延误,改善城市交通现状的理论依据。

最后,指出了我们所做工作的创新之处,介绍了模型的优缺点和改进方案。

关键词: 信号配时 平面交叉口 多目标优化 数解法 模糊控制

目录

摘要	1
一、 问题的提出	4
1.1 背景介绍	4
1.2 问题的重述	5
二、 模型假设	5
三、 符号说明	6
四、 模型的建立与求解	6
5.1 多目标动态优化配时模型	7
5.1.1 性能指标的选取	7
5.1.2 信号配时优化模型的建立	9
5.1.3 信号配时优化模型的求解	10
5.1.4 以排队长度为控制变量的模糊动态优化算法	11
5.2 两相邻交叉口信号灯实时控制模型	14
5.2.1 邻近交叉路口概述	14
5.2.2 邻近交叉路口信号灯实时控制的整体设计方案	16
5.2.3 邻近交叉路口的信号相位控制	16
5.2.4 交叉路口饱和率 $k < 1$ 时的信号灯优化模型	18
5.2.5 交叉路口饱和率 $k \geq 1$ 时的信号灯优化模型	19
5.3 车辆延误模型	21
5.3.1 相位车流在一个周期内的流量到达累计图式	21
5.3.2 某相位（某连线）车流在一个周期内的车辆延误时间	22
5.4 相位差的优化	26
5.5 配时优化算法的对比分析	29
5.5.1 时间复杂度	29
5.5.2 空间复杂度	30
5.5.3 收敛性	31
5.6 实验仿真检验与分析	31
六、 模型的评价	37
6.1 优点	37
6.2 缺点	37
七、 模型的改进	37
7.1 一种环形交叉路口的创新设计	37
八、 模型的推广	38
九、 参考文献	39
十、 附录	41

一、问题的提出

1.1 背景介绍

随着我国城市化速度的加快以及城市规模的不断扩大，交通供需矛盾日益突出，在城市交通网络中产生的交通拥堵现象日趋严重，严重影响了社会经济的发展和人民生活水平的提高。

为了提高城市道路管理水平，改善城市交通秩序，保障公路交通的畅通与安全，当今世界各国普遍使用智能交通系统。在该系统中，核心的问题是交通信号智能控制。平面交叉口是道路交通的主要冲突点，不仅机动车数量多，而且行人和非机动车也在同一平面通过。目前在我国的大、中型城市交通管理中，普遍采用的是单点定时交通信号灯控制。定时控制这种传统信号灯控制方法会造成某些方向绿时浪费，而在有些方向上车辆通行又延误严重。因此，优化交叉口信号配时是提高交叉口运行效率最有效的方法之一。

然而传统的城市交通信号灯配时控制研究的对象常常是单个交叉路口，而且往往都是利用定时控制的思想来设计交通信号灯的配时方案，有时候也考虑全天车流的显著不同把一天分割为几个时间段，然后采用分时间段的方法设计定时配时方案，或者全天候定时配时方案不变^[1]。不管哪种方案都没有考虑交通流的随机性的特点。这样的红绿灯时间及间隔都固定不变的配时方案，难以适应车流量波动明显的情况，在实际交叉路口这种配时方案常常造成交通堵塞。随着研究的一步步深入，一些大城市开始协调优化多个交叉口的交通信号灯，使相关性较大的交叉口之间形成理想的绿波带，在一定程度上使车辆通行更顺畅，减少了车辆的时滞，但这种控制方案固定不变，无法从根本上解决问题，仍然容易造成交通堵塞频发。



图 1.1 拥挤的十字路口情况

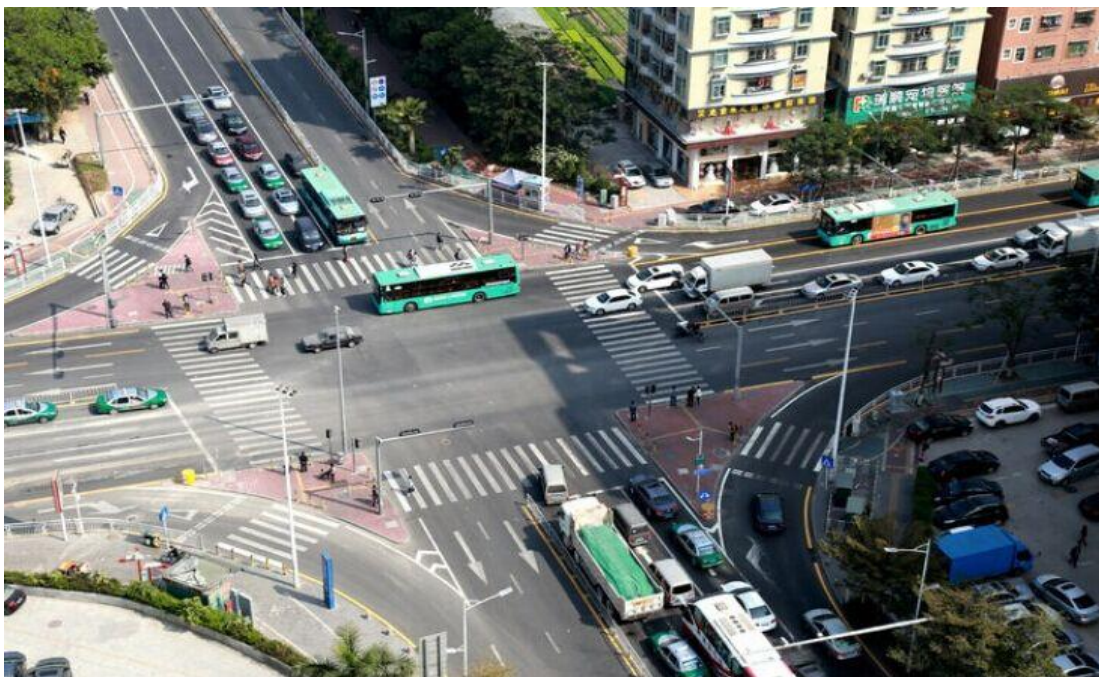


图 1.2 有秩序的十字路口情况

随着我国城市化速度的加快以及城市规模的不断扩大，交通供需矛盾日益突出，在城市交通网络中产生的交通拥堵现象日趋严重，严重影响了社会经济的发展和人民生活水平的提高。

1.2 问题的重述

为了提高城市道路交通管理水平，改善城市交通秩序，保障公路交通的畅通与安全，题目要求我们将采取一定的算法，从时间复杂度、空间复杂度、收敛性等三个方面，对交通信号进行配时优化研究，进而求解出改善后的交通信号配时方案，然后进行仿真检验，最后以期达到提高实际平面交叉口的通行能力和服务水平，减少城市交通网的交通延误，改善城市交通现状的目标。

二、 问题的分析

这是一个多目标优化设计问题，需要根据车均延误时间和停车次数等主要优化指标对交叉路口信号进行合理配时，使车均延误时间和停车次数等性能评价指标达到最优情况，从而提高平面交叉路口的通行能力和服务水平、改善城市交通秩序、保障公路交通的畅通与安全。问题的特点在于需求实验搜集数据量大，数据动态变化快，可挖掘和考虑的性能评价指标多（如延误时间、停车次数、通行能力、环境参数、燃油消耗和汽车尾气等）；难点在于，问题的解决需要大量的交通调查数据，并且交叉路口处车辆、行人等元素动态变化过于频繁和复杂，很难准确的通过模型来判别和预测车辆、行人的数量及其行为。所以我们需要在考虑多目标优化的同时动态优化配时方案，来匹配交通的实时变化。

三、 模型假设

- 车辆在道路行驶中严格遵守交通规则，且忽略交通事故对交通流和信号灯控制的影响；
- 车辆严格遵守排队过程中的先到先服务，不存在插队；
- 忽略黄灯对交叉路口交通流的影响；
- 忽略绿灯亮起时车辆启动的加速过程，车辆以空间平均速度驶出停止线；
- 车流量在非交通高峰期内服从泊松分布，在交通高峰期内服从均匀分布；
- 行人以最小通过时间穿越道路，忽略行人对直流车流的影响。

四、 符号说明

符号	符号意义
D	车辆在交叉路口的平均延误
D_0	车辆在交叉路口的初始平均延误
Q	交叉路口通行能力
Q_0	交叉口初始通行能力
H	车辆在交叉路口的平均停车次数
H_0	车辆在交叉路口的初始平均停车次数
α	平均延误的权重值， $0 \leq \alpha \leq 1$
β	通行能力的权重值， $0 \leq \beta \leq 1$
γ	平均停车次数的权重值， $0 \leq \gamma \leq 1$
g_{\min}	每相位的最小绿灯时间， $g_{\min}=15$
g_{\max}	每相位的最大绿灯时间，单位 s
L	交叉口总损失时间，单位为 s
n	交叉口相位数
C_{\min}	最小周期，单位为 s
C_{\max}	最大周期，单位为 s

五、模型的建立与求解

5.1 多目标动态优化配时模型

经过相关文献得到结论，交通状态的随机不确定性特征决定了交叉路口的交通信号灯配时优化的显著差异。在交通流趋近饱和的状态下，交通通行能力面临着最大的压力，所以应该优先减少排队长度，保证交叉路口最大限度使车流通过；而在交通流低饱和的状况下，在保证车流在交叉路口通行流畅的同时，尽可能减小时间浪费变得更具实际意义。减小车辆延误，降低停车次数都是为了提高人们出行的时间效率。交通灯配时控制策略将逐渐成为有效治堵的方法之一，我们首先建立基于多目标优化的交通灯动态配时模型，研究讨论邻近交叉口多目标的动态优化信号配时算法。

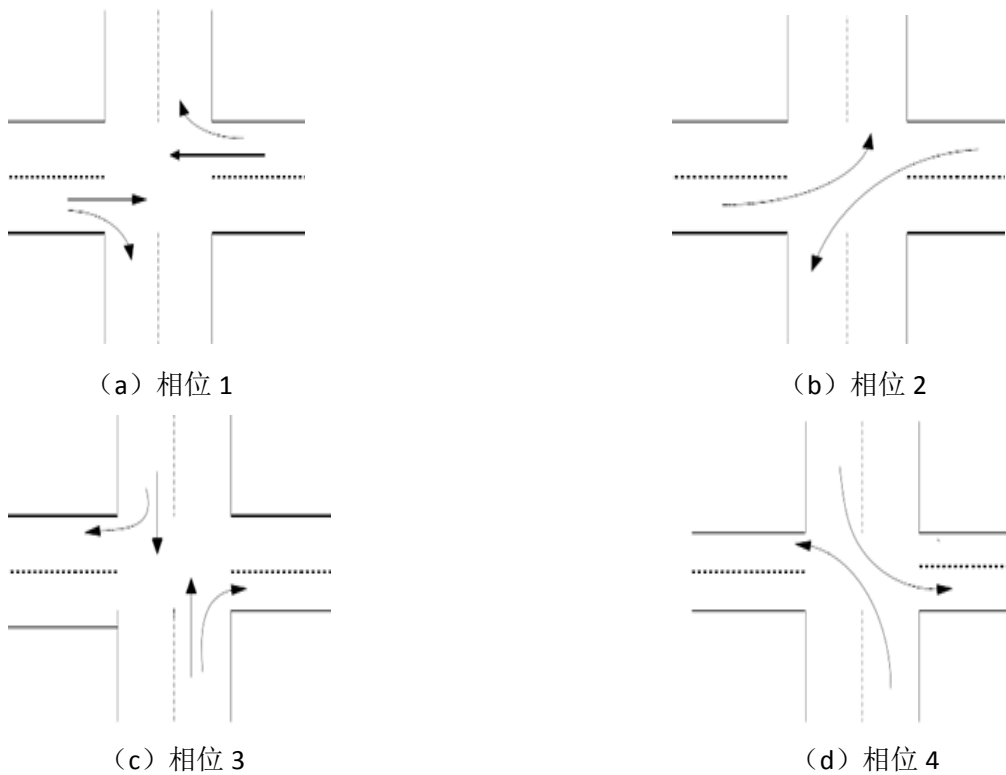


图 5.1 四个相位的设定

5.1.1 性能指标的选取

➤ 延误 d

尽可能减小进出口各相位进口道上的车辆延误是提高有限道路资源的利用效率的一个重要方法^[2]，道路上的车辆延误 d 主要由一致性延误 d_u 和随机延误 d_r 两部分组成，其中 d_u 为车辆到达率为常数的延误， d_r 为车辆到达率不一致的延误。所以第 i 相位的平均延误时间 d_i 为：

$$d_i = d_{ui} + d_{ri} = \sum_j \frac{C(1-\lambda_i)^2}{2(1-y_{ij})} + \frac{y_{ij}^2}{2\lambda_i q_{ij}(\lambda_i - y_{ij})}$$

式中： d_{ui} -- 第 i 相位的平均一致性延误，单位为 s/veh ；

d_{ri} -- 第 i 相位的平均随机延误单位为 s/veh ；

C -- 周期时长，单位为 s ；

$\lambda_i = \frac{g_{ei}}{C}$ -- 第 i 相位的绿信比；

q_{ij} -- 第 i 相位第 j 进口道的实际交通流量，由实测得到；

y_{ij} -- 第 i 相位第 j 进口道的交通强度，即实际交通流量与饱和流量之比。

设周期内车辆总延误时间 D 为：

$$D = \frac{\sum_i d_i \sum_j q_{ij}}{\sum_i \sum_j q_{ij}} = \frac{\sum_i q_i \left[\sum_j \frac{C(1-\lambda_i)^2}{2(1-y_{ij})} + \sum_j \frac{y_{ij}^2}{2\lambda_i q_{ij}(\lambda_i - y_{ij})} \right]}{\sum_i \sum_j q_{ij}}$$

➤ 排队长度

在多目标联合优化函数中我们没有加入该指标，而是把它用作模糊控制变量来实现动态追加绿灯时间，这样我们就把多目标联合优化和动态配时策略有效结合在一起。

➤ 通行能力 Q

在道路交叉口，对车辆通过交叉口的定义是车辆只有在有效绿灯时间内通过停车线^[3]。根据停车线原理，我们定义：一个车道在 1 小时内可通过停车线的车辆数即为该车道的通行能力。

$$Q_{ij} = \frac{3600}{C} \times \frac{g_{ei}}{h_{ij}}$$

式中： Q_{ij} -- 相位 i 的第 j 个车道的通行能力；

h_{ij} -- 平均车头时距。

➤ 停车次数

车辆的停车次数，又叫停车率，是车辆在通过交叉口时受车流和信号控制影响而停车的次数，是车辆在受阻情况下的停车程度^[4]。

设一个周期内交叉口的平均停车次数为：

$$H = \frac{\sum_i H_i \sum_j q_{ij}}{\sum_i \sum_j q_{ij}} = \frac{\sum_i q_i \left[\sum_j 0.9 \times \frac{C(1-\lambda_i)}{1-y_{ij}} \right]}{\sum_i \sum_j q_{ij}}$$

式中： H_i ——第 i 相位的车辆平均停车次数。

5.1.2 信号配时优化模型的建立

信号交叉口配时重要性能参数：通行能力，延误，排队长度，停车次数和饱和度等^[5]。我们暂以提高通行能力，降低延误，降低排队长度，降低停车次数这四个性能参数作为信号配时的指标。其中我们用通行能力、延误、停车次数这三个指标通过加权作为多目标联合优化的参数指标，把排队长度作为模糊控制指标，并且追加有效绿灯时间。由于单位不同，我们对联合优化的三指标进行了无量纲化处理^[6]。用不同的权值用来表达指标的重要程度。进一步，我们得到了信号配时的联合优化函数为：

$$\min PI = \alpha \frac{D}{D_0} - \beta \frac{Q}{Q_0} + \gamma \frac{H}{H_0}$$

式中： D ——车辆在交叉路口的平均延误；

D_0 ——车辆在交叉路口的初始平均延误；

Q ——交叉路口通行能力；

Q_0 ——交叉口初始通行能力；

H 车辆在交叉路口的平均停车此时；

H_0 ——车辆在交叉路口的初始平均停车次数；

α ——平均延误的权重值， $0 \leq \alpha \leq 1$ ；

β ——通行能力的权重值， $0 \leq \beta \leq 1$ ；

γ ——平均停车次数的权重值， $0 \leq \gamma \leq 1$ ；

g_{\min} ——每相位的最小绿灯时间， $g_{\min} = 15s$ ；

g_{\max} ——每相位的最大绿灯实践，单位 s ；

L ——交叉口总损失时间，单位为 s ；

n ——交叉口相位数；

C_{\min} ——最小周期，单位为 s ；

C_{\max} ——最大周期，单位为s。

在加权系数的选取上，我们考虑了交通流的实际情况，即在交通流量低饱和时尽可能减少车辆在交叉口的延误和停车，在趋近饱和和过饱和时尽可能的提高交叉口的通行能力。因为每个相位有两个方向的交通流，我们将系数乘以2。

$$a_i = 2s_i y_i (1 - Y)$$

$$\beta_i = 2 \times \frac{YC}{3600}$$

$$r_i = 2s_i y_i \frac{1 - Y}{0.9}$$

式中： s_i ——第 i 相位进口道路的饱和流量；

y_i —— 交叉口第 i 相位的流量比；

Y ——组成周期的全部信号相位的各个最大流量比 y 值之和。

5.1.3 信号配时优化模型的求解

鉴于时间限制，在综合考虑后我们决定采用“定步长搜索方法”来求解该模型^[7]，周期 C 的计算流程图如图 5.2 所示。

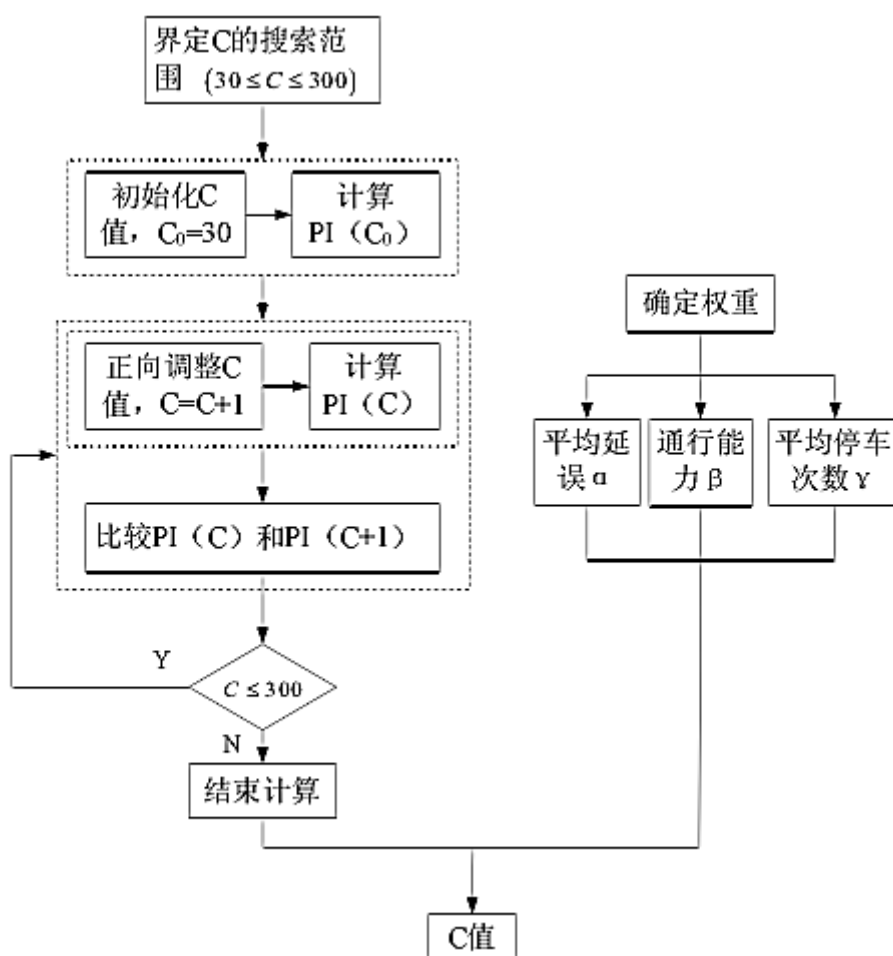


图 5.2 周期 C 的计算流程图

确定最佳周期后，总有效绿灯时间为 $G_e = C_0 - L$ ；

各相位的有效绿灯时间为
$$g_{ei} = G_e \times \frac{\max[y_i]}{Y}$$

5.1.4 以排队长度为控制变量的模糊动态优化算法

1) 模糊动态优化流程

我们以当前相位和下一相位的排队长度为模糊控制变量，按照模糊控制中双输入单输出的理论^{[8][9]}，先把当前相位的排队长度和下一相位的排队长度进行了模糊化，进一步通过模糊规则表经过运算而得到了模糊合成关系。然后我们具体以实际交叉口的当前相位和下一相位的排队长度为输入变量，由模糊控制规则得到绿灯追加时间模糊值，经过反模糊化处理最终得到了绿灯追加时间。模糊动态优化流程如图 5.3 所示。

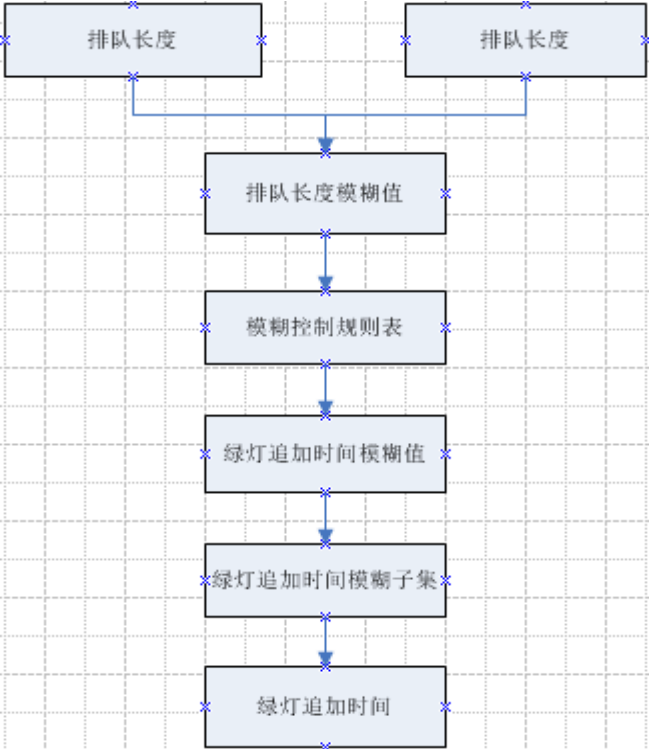


图 5.3 模糊动态优化图

2) 模糊控制规则的建立如表 5.1 所示：

表 5.1 排队长度的隶属值赋值表

语言值	排队长度							
	1	3	5	7	9	11	13	15
VS	1	0.7	0.2	0	0	0	0	0
S	0.2	0.7	1	0.7	0.2	0	0	0
M	0	0	0.1	0.8	0.8	0.1	0	0
L	0	0	0	0.2	0.7	1	0.7	0.23
VL	0	0	0	0	0	0.1	0.7	1

当前相位追加绿灯时间时的基本论域为{0,1,2.....30}，取7个语言值：VS（很短），S（短），LS（较短），M（中等），LL（较长），L（长），VL（很长）。隶属度赋值表如图 5.2 所示。

表 5.2 当前相位追加绿灯时间隶属度赋值表											
语言值	绿灯追加时间										
	1	3	5	7	9	11	13	15	17	19	21
VS	1	0.6	0.1	0	0	0	0	0	0	0	0
S	0.2	0.7	1	0.7	0.2	0	0	0	0	0	0
LS	0	0.2	0.8	1	0.8	0.2	0	0	0	0	0
M	0	0	0	0.2	0.7	1	0.7	0.2	0	0	0
LL	0	0	0	0	0	0.2	0.8	1	0.8	0.2	0
L	0	0	0	0	0	0	0	0.1	0.6	1	0.6
VL	0	0	0	0	0	0	0	0	0.1	0.6	1

确定模糊控制规则表如表 5.3 所示：

表 5.3 模糊控制规则表						
当前相位排队长度	下一相位排队长度					
	VS	S	M	L	VL	
VS	VS	VS	VS	VS	VS	
S	LS	LS	LS	S	S	
M	LL	M	M	M	M	
L	VL	VL	VL	L	LL	
VL	VL	VL	VL	VL	VL	

双输入单输出的语言控制策略由 25 条模糊条件语句构成：if L_1 and L_2 then G ，根据 Mamdani 推理方法，第 i 条模糊语句所决定的模糊关系按下式计算：

$$R_i = [L1_i \times L2_i]^{r_i} \times g_i$$

T_i 表示将矩阵 $[\bullet]$ 按行排成一系列向量。总的模糊关系 $R = \bigcup_{i=1}^{25} R_i$ 的求法是把 25 个模

糊关系进行“并”运算。已知输入为 L_1 和 L_2 ，模糊化处理采用单点模糊集合，即：

$$u(e) = \begin{cases} 1, e = e_o \\ 0, e \neq e_o \end{cases}$$

由模糊控制规则，根据模糊控制理论，通过推理运算求得输出的模糊量相位有效绿灯时间 g_i 。

公式如下所示：

$$g_i = [l1_j \times l2_j]^{r_j} \circ R$$

式中： T_2 ——将矩阵 $[\bullet]$ 按行排列成一个列向量。

计算得到如表 5.4 模糊控制绿灯追加时间查询表：

表 5.4 模糊控制绿灯追加时间查询表									
绿灯追加时间	下一相位排队长度								
	1	3		5	7	9	11	13	15
当前相位排队长度	1	4	4	4	4	4	4	4	3
	3	5	5	5	5	5	5	4	4
	5	8	8	8	7	7	7	6	6
	7	12	11	10	10	9	9	9	9
	9	15	14	13	13	13	13	13	11
	11	18	18	18	17	16	16	16	15
	13	20	20	20	18	17	17	17	18
	15	20	20	20	18	18	18	18	18

5.2 两相邻交叉口信号灯实时控制模型

在城市道路网中，大量的存在相距很近的两交叉口组，若将各交叉口分别设置为单点信号控制，则极易出现车辆经常遇到红灯，时停时开的情况，从而造成行车不畅，加重环境污染。为使车辆减少在各个交叉口上的停车时间，特别是使干道上的车辆能够畅通行使，我们考虑更具实际意义的优化方案：即将一条干道上一批相邻的交通信号连接起来，加以协调控制，也就是进行对干线交叉口交通信号的协调控制系统的研究，此系统也称为线控制或绿波系统^[10]。我们以两个单点信号控制交叉口组成的简单绿波系统为例，建立一个基于基本交通流模型以及交叉路口的排队分析理论，以两个交叉路口的车辆总延误最小为目标，以主干道的绿波效应以及中间路段的车辆排队长始终小于其最大车辆排队长度为约束条件的非线性函数模型^[11]，在对绿波系统的交叉口信号配时优化进行深入研究后发现，通过对路段和干线机动车流进行协调控制设计，可以有效地减少相交道路车流对干线车流的干扰，实现邻近交叉路口信号灯的实时控制，进一步实现在保证相交道路车流行车需求的同时，使干线车流获得最大的通行权的目标。

5.2.1 邻近交叉路口概述

邻近交叉路口是平面交叉路口的一种，由距离较近的或者交通高峰期时下游路口进车道的饱和率 $k \geq 1$ 且中间路段发生拥堵的概率较大的两个相邻交叉路口组成。而经查阅资料和相关性检验分析，我们发现，在一般情况下，当干道上相邻两个交叉口相距超过 800 米时，两个交叉口的交通灯配时相关性大大减小^[12]，我们将此种情况考虑为两个交叉路口不相关，即视为两个单点交叉路口的简单叠加，继续采用上面内容中提到的单点交叉口多目标动态优化配时模型来分析和处理此优化问题，也就是说，在多点交叉路口的信号协调优化系统中，相距 800

米以上的两相邻交叉口采用单点动态配时方案，相距距离在 800 米以内的两相邻交叉口采用信号协调优化方案。本部分我们主要针对相距 800 米以内的邻近十字路口进行研究，其交通示意图如图 5.4 所示：

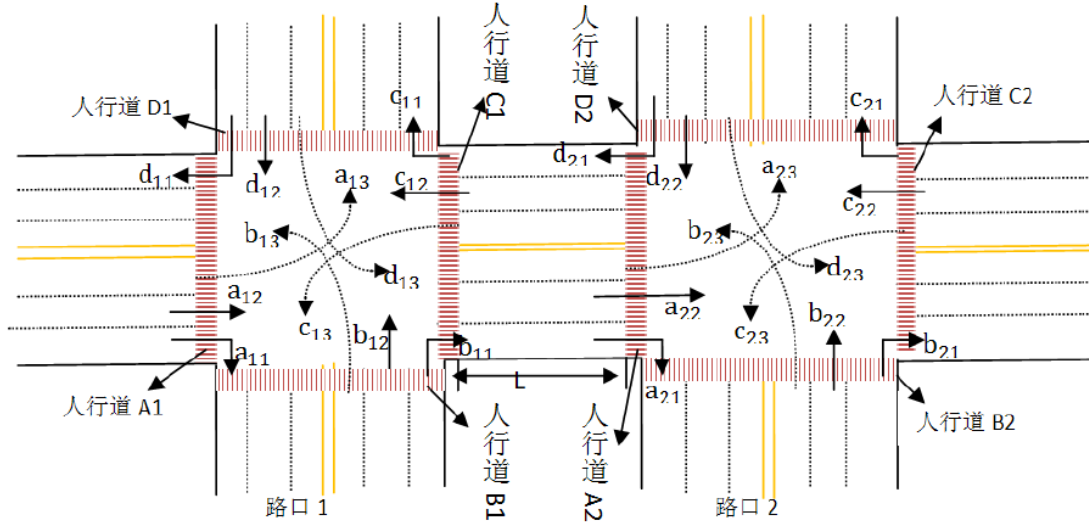


图 5.4 邻近交叉路口物理模型示意图

其中，

a_{ij} 表示车流 a 在第 i 个交叉路口的第 j 中行驶方向；

b_{ij} 表示车流 b 在第 i 个交叉路口的第 j 中行驶方向；

c_{ij} 表示车流 c 在第 i 个交叉路口的第 j 中行驶方向；

d_{ij} 表示车流 d 在第 i 个交叉路口的第 j 中行驶方向；

A_i, B_i, C_i, D_i 分别代表第 i 个交叉路口的 4 个人行横道；

L 表示邻近交叉路口之间的中间路段；

$i=1,2$ ，1 代表第一个交叉路口，2 代表第二个交叉路口；

$j=1,2,3$ ，1 代表右转，2 代表直行，3 代表左转。

邻近交叉路口是现代交通的重要组成部分，跟人们带来交通方便的同时，也带来了交通拥堵^[13]。以武汉市省政府东侧两相邻交叉路口为例，如图 5.5 所示：



图 5.5 邻近交叉路口示意图

假设武汉市省政府东侧两相邻交叉路口信号灯的的控制方式采用传统的定时控制策略，该定时控制采用直行右转同时亮起策略^[14]。由于交通高峰期时路上车流量较大，且行人通过率很大，经常产生行人与机动车抢道通过的情形，不仅对行人构成了极大的安全隐患，而且对车辆产生了较大的空间和时间上的延误，进而导致了路口上的车辆排队频率大且长度长，产生交通拥堵，甚至堵塞。

5.2.2 邻近交叉路口信号灯实时控制的整体设计方案

我们分为两层来研究邻近交叉路口信号灯的实时控制方案，如图 5.6 所示，

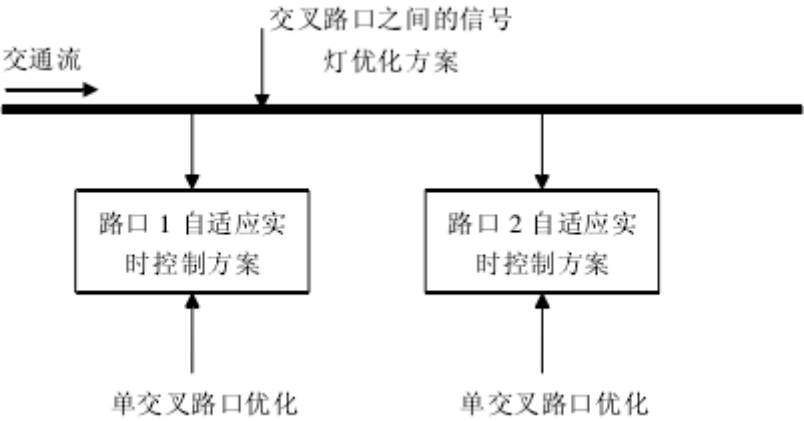


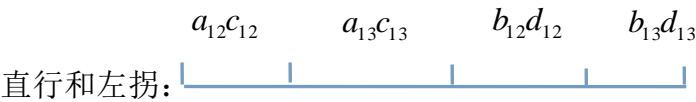
图 5.6 邻近交叉路口的整体设计方案

我们基于基本交通流模型以及交叉路口的排队分析理论，以两个交叉路口的车辆总延误为最小目标，以主干道的绿波效应以及中间路段的车辆排队长始终小于其最大车辆排队长度为约束条件建立数学模型来实现邻近交叉路口信号灯的实时控制。

我们做的第一层工作主要是基于排队论对交叉路口的信号灯实时控制的研究，以最小车辆延误为目标，实现交叉路口的交通流最优化配置；第二层工作是对邻近的两个交叉路口在同方向上的信号灯控制进行优化，此处我们主要采取设定一个绿灯时间间隔（即相位差）的方法来实现非交通高峰期时的交通流绿波效应或者交通高峰期时的中间路段排队总长度始终小于最大排队长这一目标。

5.2.3 邻近交叉路口的信号相位控制

传统的信号灯控制方案为定时控制，如图 5.7 所示，较为常见为周期 $C=90s$ 或 $120s$ 。其特点是只对直行和左拐车辆进行信号控制，对右拐车辆则不进行控制^[15]。其优点是信号控制方式简单，适用于车流量较小的交叉路口；缺点是将单位周期内绿灯时间的浪费，且人行横道、车辆右拐、直行和左拐的绿灯时间冲突，经常出现人车抢道的情景，对行人造成安全隐患，对车辆则造成了不必要的延误。



人行道: B_1D_1 A_1C_1

图 5.7 传统绿灯相位图

我们基于各路口右拐方向信号灯可以在人行道为红灯时设置为绿灯的考虑, 采用了一种新的邻近交叉路口信号控制方式, 如图 5.8 所示。

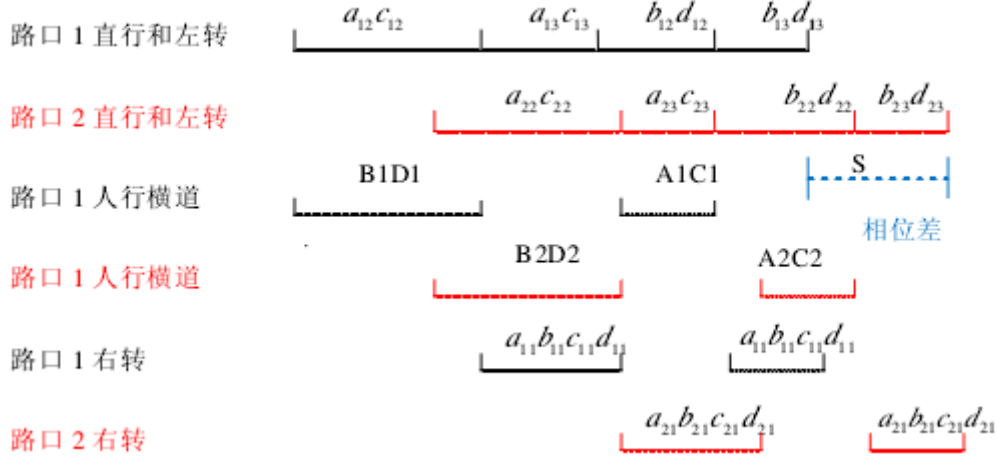


图 5.8 邻近交叉路口绿灯相位图

显然, 在该信号控制方式下, 交叉路口交通流的对称性决定了信号控制方式的对称控制, 如当 a_{12} 和 c_{12} 的相位相同时, 人行横道 B_1 和 D_1 的相位也相同。紧接着, 我们做了对信号灯的转轮的过程分析:

- ✧ 直行 a_{12} 为绿灯时, 人行道 B_1 、 D_1 和直行道 c_{12} 为绿灯;
- ✧ 当人们用 θ 时间穿越人行道 B_1 、 D_1 之后, 全部右拐车道变绿灯;
- ✧ 当 a_{12} 和 c_{12} 变红后, 左拐 a_{13} 和 c_{13} 变绿灯, 此时, 全部右拐车道变绿灯仍未绿;
- ✧ 当 a_{13} 和 c_{13} 变红灯后, b_{12} 和 d_{12} 变绿灯, 此时, 全部右拐车道仍未绿灯;
- ✧ 当过了 $(g_{b_{13}} - s)$ 时间之后, 全部右转车辆变红灯, 人行道 A_1 、 C_1 变绿灯;
- ✧ 当直行 b_{12} 、 d_{12} 变红灯后, 人行道 A_1 和 C_1 变红灯, 同时左拐 b_{13} 和 d_{13} 变绿灯。
- 由于交叉路口的对称关系, 此时完成一个信号周期。

与传统的绿灯相位图相比, 一方面, 我们此处提出的信号设计相位方案实现了邻近交叉路口的两个信号灯的实时控制, 也就是设计了一种可变的信号灯周期和信号灯各相位的实时变化, 这样一来, 交叉路口的信号灯可以自适应实时交通流的各时间段最小车辆延误, 从而降低了车辆拥堵的发生概率或缩短了拥堵的持续时间; 另一方面, 我们将人行道绿灯时间与右转车辆绿灯时间隔离, 在充分保证行人穿越马路安全性的情况下, 使得中间路段的等待车辆始终保持在其所容量的最大等待车辆数之内, 从而降低了交通瘫痪的发生概率, 促使城市交通井然有序。

5.2.4 交叉路口饱和度 $k < 1$ 时的信号灯优化模型

- 相邻交叉路口的绿波效应

为了使交叉路口的车辆总烟雾最小^[16]，基于模型的假设，我们提倡交叉路口的信号灯存在一个相位差 s (即同方向上，上游路口的绿灯实践点比下游路口的绿灯时间点提前 s) 以保证上游车辆达到下游路口时，下游路口同方向的信号灯尽可能处于绿灯状态，从而减少停车次数，降低车辆总延误。因此，假设在上游路口车道的信号灯为绿灯时，车辆 M 以时间平均速度驶出的停车线并驶向中间路段 L ，当车辆 M 恰好到达停车线时，信号灯为绿灯的条件是：

$$\frac{L}{u_t} = nC_2 + s$$

式中

$$n = \left\lceil \frac{L}{C_2 u_t} \right\rceil$$

- 信号灯优化模型

(1) 以两个交叉路口的车辆总延误最小为目标函数，即：

$$\text{Obj. min } TD = \sum_k \sum_i \sum_j \frac{Q_{Mkij} t_{Qkij}}{2}$$

(2) 约束条件的确定

根据交通经验可得：

$$\begin{aligned} g_{kij\min} &\leq g_{kij} \leq g_{kij\max} \\ C_{i\min} &\leq C_i \leq C_{i\max} \\ 0 &< \bar{u}_t \leq u_{\max} \end{aligned}$$

根据交叉路口的信号周期定义：

$$\sum_{k=a,b} \sum_{j=2,3} g_{kij} = C_i$$

交叉路口的信号相位对称性约束：

$$\begin{aligned} g_{aij} &= g_{cij} \\ g_{bij} &= g_{dij} \end{aligned}$$

交通流的绿波效应：

$$\frac{L}{u_t} = nC_2 + s$$

交叉路口饱和度小于 1：

$$\begin{aligned} \frac{\lambda_{kij}}{\mu_{kij}} &< 1 \\ n &= \left\lceil \frac{L}{C_2 u_t} \right\rceil \end{aligned}$$

式中,

$k = a, b, c, d$ 代表交叉路口的四个分路口。

$i = 1, 2$; 1 代表第一个交叉路口, 2 代表第二个交叉路口;

$j = 1, 2, 3$; 1 代表右转; 2 代表直行, 3 代表左转;

TD : 车辆总延误;

C_i : 第 i 个交叉路口的信号灯周期;

s : 邻近交叉路口的相位差;

g_{kij} : 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的绿灯时间;

Q_{mkij} : 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的排队车辆数;

t_{Qkij} : 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的最大车辆等待时间;

u_i : 中间路段向右行驶方向的时间平均速度;

L : 中间路段长度。

5.2.5 交叉路口饱和率 $k \geq 1$ 时的信号灯优化模型

● 高峰期时中间路段的畅通性约束

当交叉路口处于过饱和状态, 即交叉路口饱和率 $k \geq 1$ 且持续时间较长时, 则可能出现这样一种情况: 交叉路口的车辆到达率较大, 中间路段的车辆排队长度也很大, 且上游路口中驶入中间路段的各相位信号灯依旧处于绿灯的情况下, 更多的车辆会选择通行并驶入中间路段, 从而造成中间路段的交通拥堵或者堵塞^[17]。因此, 为了保持中间路段的交通畅通性, 应尽可能的使得每一个交通周期内, 中间路段的车辆排队长始终等于或者小于中间路段的最大车辆数, 其公式为:

$$Q_M = \sum Q_{\lambda} - \sum Q_{\text{出}} \leq Q_{\text{max}}$$

式中 Q_M ——单位周期内中间路段的排队车辆数;

Q_{λ} ——单位周期内中间路段的驶入车辆数;

$Q_{\text{出}}$ ——单位周期内中间路段的驶出车辆数;

因此, 当交通高峰期时上游路口的第一个信号周期结束时, 中间路段所停留的车辆数为:

$$Q_{M1} = (\lambda_1 - \mu)C_1$$

式中, Q_{M1} ——交通高峰时上游路口的第一个信号周期结束时的排队车辆数;

λ_1 ——交通高峰时上游路口的第一个信号周期内中间路段车辆到达率;

μ ——下游路口 a_2 的车辆服务率;

当高峰期的第 x 个周期结束时, 中间路段所停留的车辆数为:

$$Q_{mx} = \sum [(\lambda_x - \mu)C_1]$$

式中, Q_{mx} ——第 x 个交通高峰期时的中间路段的排队车辆数;

λ_x ——中间路段的车辆到达率;

μ ——路口 a_2 中各车道的车辆服务率;

C_1 ——第 x 个交通高峰时期时上游路口的信号周期。

车辆在道路的行驶中守恒^[18]，且每个路口均有多条车道组成，因此，上游路口驶入中间路段的车辆等于下游路口 a_2 的车辆到达率。为了便于计算，将中间路段的到达率转化为上游路口的驶出率，即：

$$Q_{Mx} = \sum \frac{g_{kij} \mu_{kij}}{C_1} - \sum \frac{g_{a2j} \mu_{a2j}}{C_2}$$

式中：

g_{kij} ——交通高峰期时的上游路口的第 x 个交通周期结束时上游路口中行驶方向为中间路段的各车道绿灯时间；

u_{kij} ——交通高峰期时的上游路口的第 x 个交通周期结束时上游路口中行驶方向为中间路段的各车道服务率；

g_{a2j} ——交通高峰期时的上游路口的第 x 个交通周期结束时下游路口 a_2 的各车道绿灯时间；

μ_{a2j} ——交通高峰期时上游路口的第 x 个交通周期结束时下游路口 a_2 的个车道服务率；

由于整个交通过程中，都必须保证中间路段的排队车辆数小于最大排队车辆数，所以，

$$\sum \frac{g_{kij} \mu_{kij}}{C_1} - \sum \frac{g_{a2j} \mu_{a2j}}{C_2} \leq Q_{\max}$$

交叉路口饱和度大于或者等于 1：

$$\frac{\lambda_{kij}}{\mu_{kij} \frac{g_{kij}}{C_i}} \geq 1$$

$$n = \left\lceil \frac{L}{C_2 u_t} \right\rceil$$

式中，

$k = a, b, c, d$ 代表交叉路口的四个分路口；

$i = 1, 2$ ；1 代表第一个交叉路口，2 代表第二个交叉路口；

$j = 1, 2, 3$ ；1 代表右转；2 代表直行，3 代表左转；

TD ：车辆总延误；

C_i ：第 i 个交叉路口的信号灯周期；

s ：邻近交叉路口的相位差；

g_{kij} 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的绿灯时间；

Q_{mkij} 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的排队车辆数;

λ_{kij} : 车辆 k 在交通路口高峰期的第 x 个交通周期内地 i 个交叉路口的第 j 种行驶方向上的车辆到达数;

u_{kij} : 车流 k 在交通高峰期的第 x 个交通周期内的第 i 个交叉路口的第 j 种行驶方向上的车辆服务率;

t_{Qkij} 车流 k 在第 i 个交叉路口的第 j 种行驶方向上的最大车辆等待时间

g_{a2j} : 交通高峰时上游路口第 x 个交通周期结束时下游路口 a_2 各车道绿灯时间;

u_{a2j} : 交通高峰期时上游路口第 x 个交通周期结束时下游路口 a_2 各车道服务率;

u_t : 中间路段向右行驶方向的时间平均速度;

L : 中间路段长度

● λ 值的确定

由于交通流的不稳定状态, λ 的值处于不断的变化当中, 而我们所期望的值是一个定值, 于是, 我们采用求和平均化处理, 即:

$$\lambda = \frac{\sum \lambda_x}{x}$$

式中, λ_x 为第 x 个交通流状态的车辆到达率;

5.3 车辆延误模型

为了能更好的量度平面十字交叉路口的服务能力, 我们又进一步改进以上模型, 建立了车辆延误模型, 系统的分析车辆在交叉路口的延误情况, 进一步对模型优化, 提高模型的健壮性。

5.3.1 相位车流在一个周期内的流量到达累计图式

车辆累计到达图式存在两种情况, 第一种情况是红灯期间到达的车辆在绿灯时间内释放完, 第二种情况是红灯期间到达的车辆在绿灯时间内未释放完。

如图 5.3.1 所示:

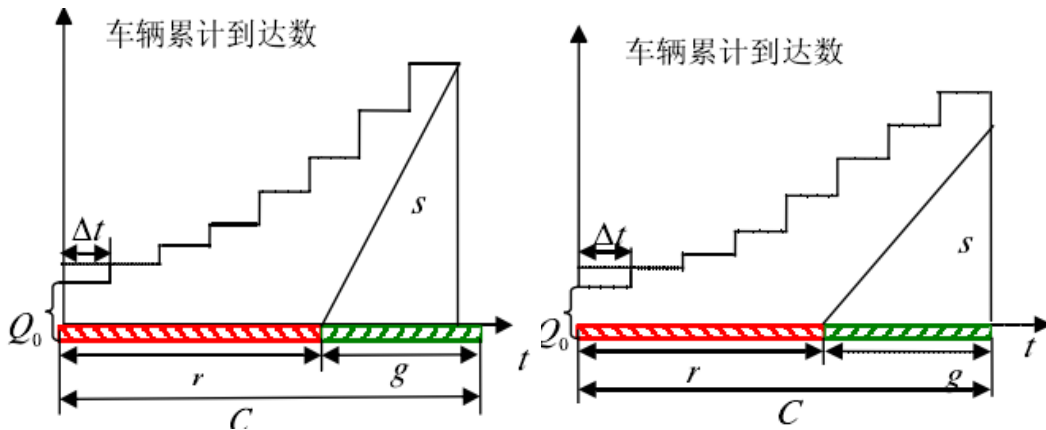


图 5.3.1 车辆延误图示

各个时间段 Δt 内连线到达流量为了避免数据的随机性，要求检测器连续上报了多个周期的各个 Δt 的车辆到达数，因此一个周期内各个时间段 Δt 内的车辆到达数为同期的车辆到达的均值。

$$q_{ji} = \frac{q_{ji}^1 + q_{ji}^2 + \dots + q_{ji}^k}{k}$$

式中：
 q_{ij}^{--j} 连线在第 i 个 Δt 内的车辆到达数 (pcu/h)；
 q_{ij}^{k--j} 连线在第 i 个 Δt 内的第 k 次车辆到达数 (pcu/h)。

初始累计流量 N_{oi}^j 的计算模型

$$N_{0ji} = \sum_0^i q_{ji}$$

式中：
 N_{0ji}^{--j} 连线初始的在第 i 个 Δt 内的累计流量 (pcu/h)；
 q_{ji}^{--j} 连线的在第 i 个 Δt 内的到达流量 (pcu/h)；

5.3.2 某相位（某连线）车流在一个周期内的车辆延误时间

交叉口的延误一般由正常阻滞延误和平均饱和延误两种延误形式组成，即：

$$d_i = d_{li} + d_{2i}$$

式中：
 d_i^{--i} 相位车辆的平均延误时间 (s)；
 d_{li}^{--i} 相位车辆的正常阻滞延误时间 (s)；
 d_{2i}^{--} 平均过饱和车辆延误时间 (s)。

1) 平均过饱和车辆延误时间 d_{2i} 的计算方法

$$d_{2i} = \frac{m_i x_i}{q_i}$$

式中：
 d_{2i}^{--} 平均过饱和车辆延误时间 (s)；
 x_i^{--i} 相位的饱和度， $x_i = y_i / \lambda_i$ ；
 y_i^{--i} 相位的流量比， $y_i = q_i / S_i$ ；
 q_i^{--i} 相位的关键车道的流速 (pcu/h)；
 S_i^{--i} 相位的关键车道的饱和流率 (pcu/h)；

q_i -- i 相位的流率 (pcu/h), $q_i = \sum_{j=1}^k q_{ij}$;

q_{ij} -- i 相位的 j 车道的流量 (s);

k -- i 相位中车道总数;

λ_i -- i 相位的绿信比, $\lambda_i = g_i / C$;

g_i -- i 相位的绿灯时间 (s);

C -- 各交叉口的周期时长 (s).

$$m_i = \begin{cases} \frac{\frac{Q_i t}{4} (Z_i + \sqrt{Z_i^2 + \frac{12(x_i - x_{i0})}{Q_i t}})}{0}, & x_i > x_{i0} \\ 0, & x_i \leq x_{i0} \end{cases}$$

m_i -- 平均过饱和滞留车队长度, 用平均排队长度车辆总数表示, 其中包括车辆

到达率随机波动构成的排队长度 (pcu/h);

$Q_i t$ -- 在时间段 t 内 i 相位可以被放行交叉口的极限车辆数 (pcu/h);

C -- 交叉口的周期时长 (s);

Q_i -- i 相位的允许通行能力 (pcu/h)。

$Z_i = x_i - 1$

$$x_{i0} = 0.67 + \frac{S_i g_i}{600 * 3600}$$

式中: S_i -- i 相应的饱和流率 (pcu/h);

g_i -- i 相位的绿灯时间 (s)。

$$S_i = \sum_{j=1}^k S'_{ij}$$

式中: S'_{ij} -- i 相位 j 车道的饱和流率 (pcu/h);

k -- i 相位中车道总数。

2) 正常阻滞延误 d_{li} 的计算模型

有了车辆到达率图式,便不难计算每周期车辆通过交叉口时的受阻延时时间。我们假定在一定的时间段内,车辆的到达率图式不发生变化,以此为基础计算车辆的延误时间,即“正常阻滞延误时间”。累计流量到达图中“锯齿”线代表车辆的累计到达量,斜直线代表最大放行率(饱和流率)。他们之间的水平坐标差即为某一瞬时到达的车辆的受阻延误时间,而纵坐标差即为瞬时的排队长度。显然这两条线与横坐标周围成的面积即为全部车辆延误时间,这一数值除以到达车辆总数,即为一周内每辆车的平均延误时间。若将结果再除以周期长度,便得到单位时间每辆车的平均延误时间。

(1) 如果绿灯期间排队车辆能完全消散($3600\frac{N}{S} \leq g_e$),如图 车流到达图式一 所示:

$$\text{则延误模型为: } D_i = \sum_1^n N_j \Delta t - \frac{1800N^2}{S_i}$$

式中: D_i -- i 相位的总延误(s);

N_j -- 第 j 时段累计车辆到达数(pcu/h);

Δt -- 流量数据的统计间隔时间(s);

N -- 最大排队车辆总到达数(pcu/h);

S_i -- i 相位饱和流率(pcu/h);

n -- 车队释放完毕统计间隔数;

$$n = \text{int}\left(\frac{3600\frac{N}{S} + r}{\Delta t}\right)$$

r -- 相位红灯时长(s)。

(2) 如果绿灯期间排队车辆不能完全消散,如图 车流到达图式一 所示:则延误模型为:

$$D_j = \sum_1^n N_i \Delta t - \frac{Sg_e^2}{2} - (n' \Delta t - C)N'$$

式中: g_e -- 有效绿灯时间(s);

C -- 周期时长(s);

N' -- 一个周期车辆的累计车辆数目(pcu/h);

n' -- 车队释放完毕统计间隔数。

$$n' = \text{int} \frac{C}{\Delta t}$$

平均阻滞延误 d_{li} 计算:

$$d_i = \frac{D_i}{N_i C}$$

式中: d_i -- 车辆正常阻滞的平均延误 (s);

D_i -- i 相位的总延时时间 (s);

N_i -- i 相位到达车辆数 (pcu/h)。

(3) 车辆总延误

相位车辆总延误时间 D'_i :

$$D'_i = (d_{li} + d_{2i}) N_i$$

式中: D'_i -- i 相位的总延时时间 (s);

d_{li} -- i 相位车辆的正常阻滞延误时间 (s);

d_{2i} -- i 相位车辆平均过饱和车辆延误时间 (s);

N_i -- i 相位到达车辆数 (pcu/h)。

5.4 相位差的优化

为了能更好的使两相邻的交叉路口协调配合工作, 我们采用数解法^{[19][20]}以最大绿波带为优化目标通过寻找使得系统中各实际信号距理想信号的最大挪移量最小来对相位差进行了优化设计。为能使模型具有更好的通用性, 我们考虑了 6 个平面交叉口的情况来进行相位差的优化设计, 最后再取一特殊值用于本题中相邻两个平面交叉路口的相位差优化设计。

设有 A, B, \dots, F , 共 6 个交叉口, 它们相邻间距列于表 5.5 第二行中, A, B 交叉口之间距离为 360m, B, C 为 500m 等, 取有效数字 36、50……。通过单点信号控制方法计算得到关键交叉口周期时长为 80 s, 相应的车辆行驶速度为 12.5 m/s。

(1) 计算 a 列

我们根据系统的公用周期和车队速度计算得到交叉口的理想间距是 500m ($VC/2 = 12.5 \times 80 = 500\text{m}$, 取有效数字 50)。这就是说, 以 A 为起始信号, 则其下游同 A 相距 $VC/2$ 、 VC 、 $VC/2$ ……处即为正好能组成交互式或同步式协调

控制的“理想信号”位置。考察下游各实际信号位置同各理想信号错移的距离，显然，此错移距离越小则信号协调效果越好。然后将 $VC/2$ 的数值在实用允许的范围内变动，逐一计算寻求协调效果最好的各理想信号的位置，以求得实际信号间协调效果最好的双向相位差。以 50 ± 5 作为最适当的 $VC/2$ 的变动范围，即 $45 \sim 55$ ，将此范围填入表 5.5 的 a 列， a 列代表交叉口的理想间距

表 5.5 数解法确定信号相位差

间距 a	36	50	25	40	60	b
45	36	41	21	16	31	16
46	36	40	19	13	27	13
47	36	39	17	10	23	13
48	36	38	15	7	19	17
49	36	37	13	4	15	21
50	36	36	11	1	11	25
51	36	35	9	49	7	26
52	36	34	7	47	3	27
53	36	33	5	45	52	28
54	36	32	3	43	49	29
55	36	31	1	41	46	30

(2) 计算 a 列内各行

我们以 $a = 45$ 行为例，画一横轴，按比例标上 $A \sim F$ 各交叉口及其间距。在图 5.9 中，从 A 点出发画等间距的直线段。例如 $ab = bc = cd = de = ef = 45$ ，从图中查出各个交叉口与前一个理想交叉口的末端的距离间隔，填入表 5.5 中相应的。

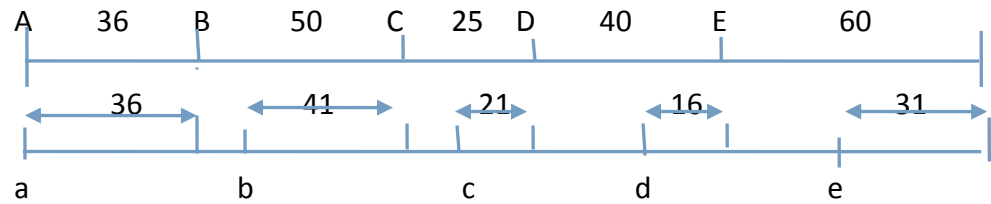


图 5.9 用数解法求表 5.5 中的数值

(3) 计算 b 列

我们仍以 $a = 45$ 行为例，将实际信号位置与理想信号的挪移量，按从小到大的顺序排列，并计算各相邻挪移量之差，将此差值最大者记入 b 列。 $a = 45$ 一行的 b 值为 16。计算方法如表 5.6 所示：

表 5.6 数解法中挪移量最大的确定表

A	E	D	F	B	C	A
0	16	21	31	36	41	45
16	5	10	5	5	4	

我们依此类推，计算 $a=35 \sim 54$ 各行之 b 值。

(4) 确定最合适的理想信号位置

由表 5.5 中可知，当 $a=55$ ， $b=30$ 时， $A \sim F$ 各信号到理想信号的相对挪移量最小，即当 $VC/2=550m$ 时可以得到最好的系统协调效率。如图 5.10 所示，图上 $A \sim E$ 同理想信号间的挪移量之差最大，则理想信号同 E 间的挪移量为 30，即各实际信号距理想信号的挪移量最大为 12.5。即各实际信号距理想信号的挪移量最大为 12.5。

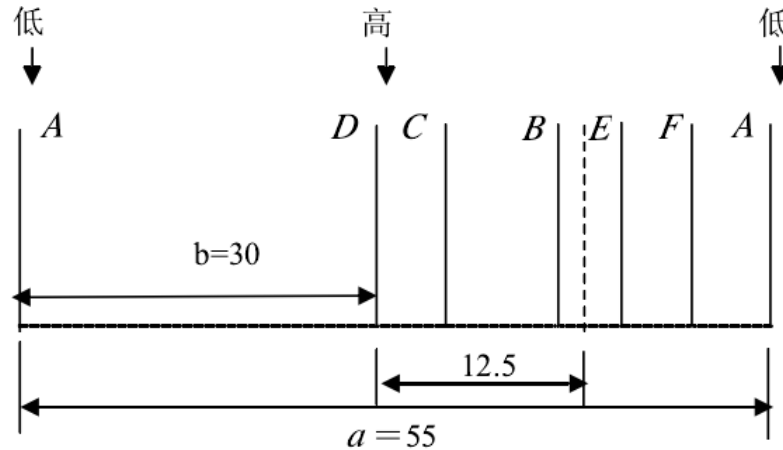


图 5.10 理想信号位置

理想信号距 D 为 125m，则距 A 为 115m，即自 A 前移 115m 即为第一理想信号，然后按次每 550m 间距将理想信号列在实际信号间，如图 5.11 所示。

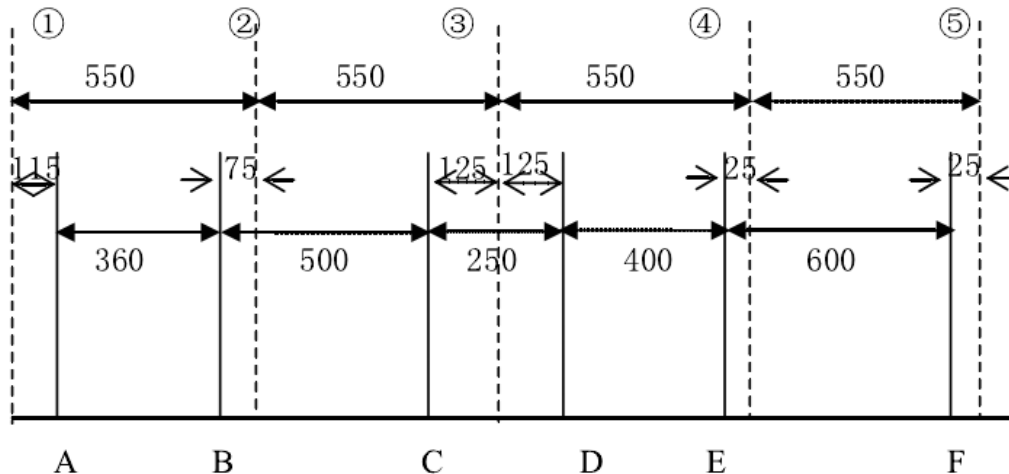


图 5.11 理想信号与实际信号的相对位置

(5) 作连续行驶通过带

在图 5.11 中我们把理想信号按次列在最靠近的实际信号下面（表 5.7 第 2 行），再把各信号（ $A \sim F$ ）在理想信号点左右位置填入表 5.7 第 3 行。把各交叉信号配时计算所得的主干道绿信比^[21]（以周期的%计）列入表 5.7 第 4 行。因实际信号与理想信号位置不一致所造成的绿时损失（%）以其位置挪移量除以理想

信号的间距（即 $a=550$ ）表示，如 A 交叉口的绿灯损失为 $115/550=21\%$ ，我们将其列入表 5.7 第 5 行。

从各交叉口的绿信比减去绿时损失即为各交叉口的有效绿信比^[22]，列入表 5.7 第 6 行，则连续通过带的带宽为左、右两端有效绿信比最小值的平均值。从表 5.7 中可知，连续通过带带宽为 A 交叉口的有效绿信比 34% 与 B 交叉口的有效绿信比 46% 的平均值 40%。

（6）求相位差

从图 5.11 及表 5.7 可见，合用一个理想信号点的左右相邻实际信号间，该用同步式协调；其它各实际信号间都用交互式协调，因此，每隔一个理想信号的实际信号间又是同步式协调。此例中，凡奇数理想信号相应的实际信号为同步协调；而偶数理想信号相应的实际信号间为交互协调。于是，相应于奇数理想信号的实际信号的相位差为 $(1 - 0.5\lambda)C$ ；相应于偶数理想信号的实际信号的相位差为 $(0.5 - 0.5\lambda)C$ 。表 5.7 第 7 行为求得的相位差值。如保持原定周期时长,则系统带速需调整为：

表 5.7 相位差计算结果

交叉路口 理想信号编号	A	B	C	D	E	F
	①	②	③	③	④	⑤
各信号位置	右	左	左	右	左	右
绿信比 (%)	55	60	65	65	60	65
损失 (%)	21	14	5	23	5	5
有效绿信比 (%)	34	46	60	42	55	60
相位差 (%)	72.5	20.0	67, 5	67.5	20.0	67.5

5.5 配时优化算法的对比分析

5.5.1 时间复杂度

我们考虑交叉路口在每天的不同时间段会承担不同的服务压力，如上下班高峰期交叉口的通行压力会远远大于深夜等车流量较少的时间段，于是，我们基于交叉路口的时间复杂度，以当前相位和下一相位的排队长度为模糊控制变量，按照模糊控制中双输入单输出的理论，先把当前相位的排队长度和下一相位的排队长度进行了模糊化，进一步通过模糊规则表经过运算而得到了模糊合成关系。然后我们具体以实际交叉口的当前相位和下一相位的排队长度为输入变量，由模糊控制规则得到绿灯追加时间模糊值，经过反模糊化处理后最终得到了绿灯追加时间。建模求解及分析过程在“5.1.4 以排队长度为控制变量的模糊动态优化算法”部分已经详细阐述，此处仅仅给出相应的动态优化流程和绿灯追加时间查询表^[22]，具体内容我们不再进行赘述。模糊控制绿灯追加时间查询表如表 5.8 所示，模糊动态优化流程如图 5.13 所示：

表 5.8 模糊控制绿灯追加时间查询表									
绿灯追加时间	下一相位排队长度								
	1	3		5	7	9	11	13	15
当前相位排队长度	1	4	4	4	4	4	4	4	3
	3	5	5	5	5	5	5	4	4
	5	8	8	8	7	7	7	6	6
	7	12	11	10	10	9	9	9	9
	9	15	14	13	13	13	13	13	11
	11	18	18	18	17	16	16	16	15
	13	20	20	20	18	17	17	17	18
	15	20	20	20	18	18	18	18	18

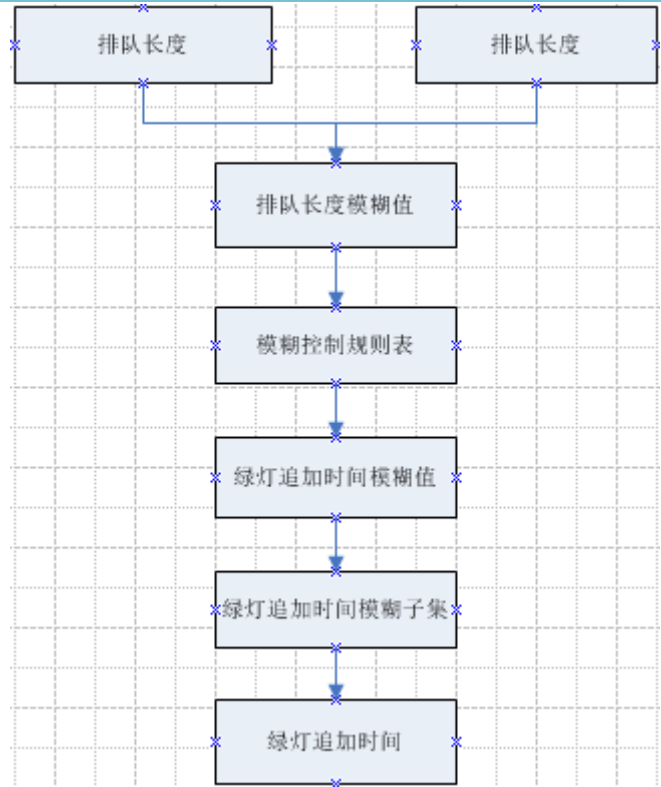


图 5.13 模糊动态优化流程图

5.5.2 空间复杂度

考虑交叉路口需要承担重型车辆、公交车、轿车、非机动车、行人等不同元素通行任务，并且需面临拥挤和相对畅通等不同的公路饱和程度^[23]，如图 5.13：



图 5.13 交叉路口的复杂情况展示

我们从空间复杂度上入手考察了交叉路口饱和率 k 分别在 $k \geq 1$ 和 $k < 1$ 不同情况下交叉路口的服务水平和通行能力，并且分别建立了交叉路口饱和率 $k < 1$ 和 $k \geq 1$ 时的信号灯优化模型，以此来进行对交通信号进行配时优化的研究。

至于模型建立及其求解的具体研究内容，在 5.2.4 和 5.2.5 部分已给出详尽的阐述，此处亦不进行赘述。

5.5.3 收敛性

对于遗传算法来说，收敛性具有重要的理论和现实意义，它是衡量短发极限时间动态行为的重要依据^[24]。设、分别为遗传算法的第 t 秒交通的最大车流量和全局最优车流量的适应值，则遗传算法的集中收敛性如下所示：

(1) 完全收敛：对任意的 $\varepsilon > 0$ ，

$$\lim_{t \rightarrow \infty} \sum_{i=0}^t P(|F_i - f^*| > \varepsilon) < \infty$$

(2) 几乎必然收敛：

$$p(\lim_{t \rightarrow \infty} |F_t - f^*| = 0) = 1$$

(3) 依概率收敛：对任意的 $\varepsilon > 0$ ，

$$\lim_{t \rightarrow \infty} P(|F_t - f^*| > \varepsilon) = 0$$

依均值收敛：

$$\lim_{t \rightarrow \infty} E[|F_t - f^*|] = 0$$

5.6 实验仿真检验与分析

由于本文将武汉市邻近的两个交叉路口 A 和 B 作为整体进行考虑，因此，只需考虑整体系统的输入与输出车流。本文通过数据统计，该数据组包含了交通非

高峰期到高峰期主要的多种交通流状态：相同车辆在不同的车道上的饱和率不同，高峰期的来临、延续与消褪等等。然后信号灯的控制策略采用图 5.14 的控制方案，采用本文的优化模型进行 *MATLAB* 仿真实验。记武汉市 A 交叉口为上游交叉路口，其数据如表 5.9 所示：

表 5.9 上游路口实验数据

车 道		a_i			b_i			d_i		
		左	直	右	左	直	右	左	直	右
1	到达率	223	353	80	118	245	123	120	202	123
	饱和率	0.07	0.12	0.06	0.09	0.18	0.12	0.09	0.09	0.09
2	到达率	216	364	86	128	256	134	113	234	106
	饱和率	0.06	0.12	0.06	0.09	0.19	0.12	0.08	0.11	0.08
3	到达率	245	433	91	120	253	120	127	246	120
	饱和率	0.08	0.14	0.06	0.09	0.18	0.12	0.09	0.12	0.09
4	到达率	360	469	136	189	366	164	244	489	211
	饱和率	0.11	0.15	0.09	0.12	0.24	0.16	0.18	0.24	0.16
5	到达率	465	962	365	196	375	161	246	493	245
	饱和率	0.16	0.32	0.19	0.13	0.24	0.16	0.18	0.24	0.18
6	到达率	723	1069	480	261	480	240	346	706	480
	饱和率	0.22	0.35	0.24	0.19	0.32	0.24	0.23	0.35	0.36
7	到达率	789	1329	547	367	567	470	569	936	703
	饱和率	0.26	0.44	0.27	0.24	0.37	0.48	0.34	0.47	0.53

记武汉市 B 交叉口为下游交叉路口，其数据如表 5.10 所示：

表 5.10 下游路口实验数据

车 道	a_1			b_1			a_2			b_2		
	左	直	右	左	直	右	左	直	右	左	直	右
1	19	30	39	18	22	39	21	30	40	19	20	40
2	19	30	40	18	23	40	21	31	45	19	20	41
3	20	35	44	20	24	44	22	35	51	20	24	51
4	22	36	48	21	23	48	23	36	53	20	24	53
5	26	37	60	20	27	60	25	37	62	22	28	62
6	28	39	72	26	29	72	29	40	73	24	30	73
7	30	43	72	25	30	78	30	43	78	25	30	78
8	32	43	80	25	29	79	31	44	80	25	30	80
9	31	44	81	26	30	81	31	44	80	25	30	80
10	32	44	83	26	31	83	32	43	81	24	32	81
11	29	41	72	24	28	72	29	39	70	24	28	70
12	23	37	65	21	24	65	21	36	64	20	27	64

对于上游路口 $C1$ 的交通流数据可以通过一个周期内的下游路口的驶入车辆进行计算,其车辆在路口 $C1$ 的行驶方向比例可以参照上一个周期的下游路口 $C2$ 比如 $C1$ 在状态 2 时的车辆行驶方向左转、直行、右转比例大约为 1:2: 1.下游路口 a_2 的交通流状态可以采用相同方案进行确定。

由表 5.9 和表 5.10 分析可知,下游交叉路口的实验数据在表 5.9 的基础上延迟一个周期,主要是因为交通流存在一定的延迟性^[24];状态 1 到状态 7 的过程是交通流逐渐增大的交通非高峰期,该阶段内,车辆密度逐渐增大,车速随之降低,车辆的产生服从或者近似服从泊松分布^[25],车辆可以在道路上自由行驶,交叉路口的饱和率较低;由状态 8 到状态 11 是交通高峰期的持续期,该阶段内车辆只能以道路平均速度跟随前车进行行驶,交通密度接近最大,平均车速降至最低,交叉路口饱和率逐渐达到最大;12 状态是交通高峰的消退期,该阶段内,车辆密度逐渐减小,车速逐渐增加,等待车辆逐渐减少,交通状态趋向正常化,例如,路口 a_1 的各车道车辆在整个交通过程中各车道的车速变化,如图 5.14 所示,

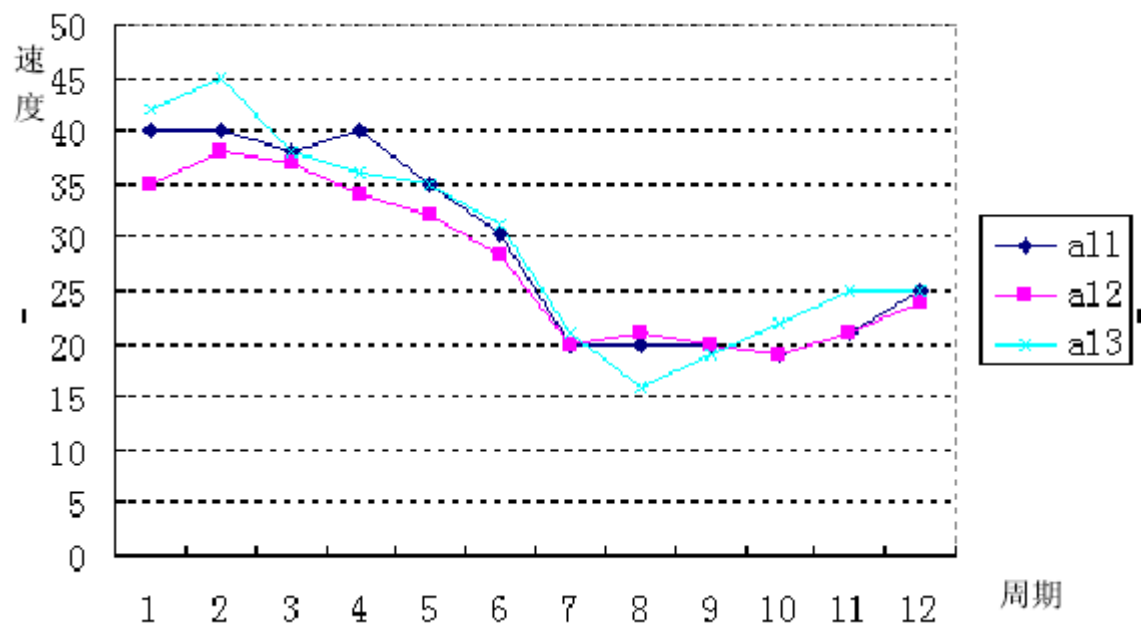


图 5.14 路口 a1 的各车道的车速

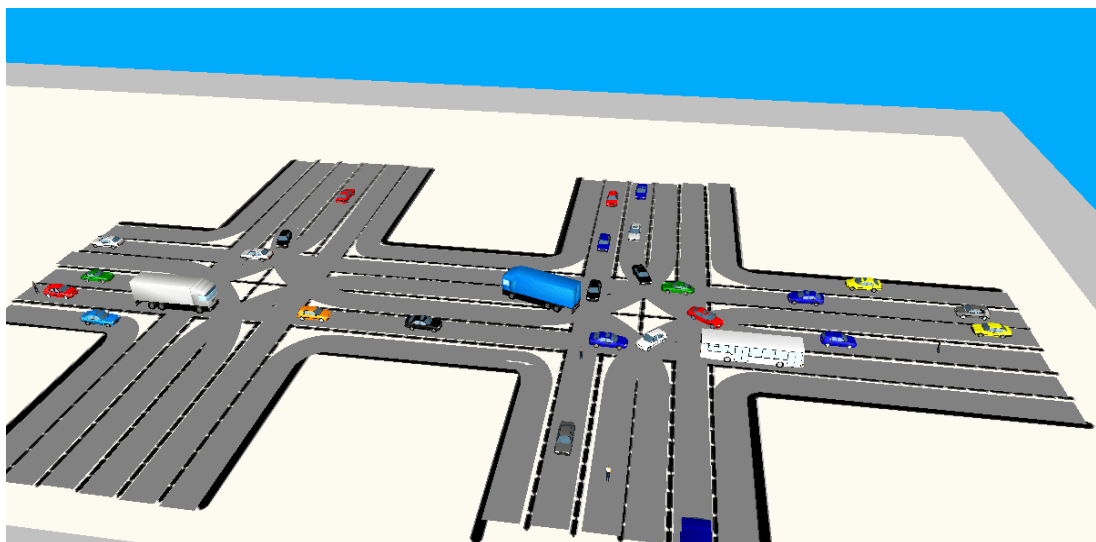


图 5.15 仿真效果图

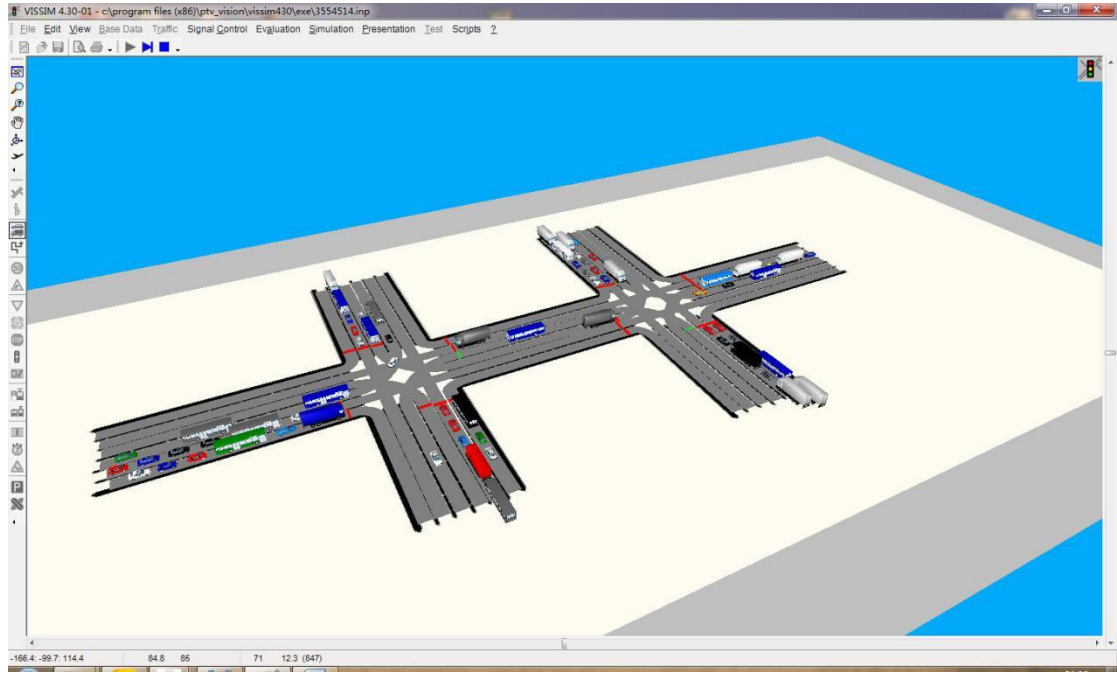


图 5.16 仿真效果图

车辆总延误对比，如图 5.17 所示：

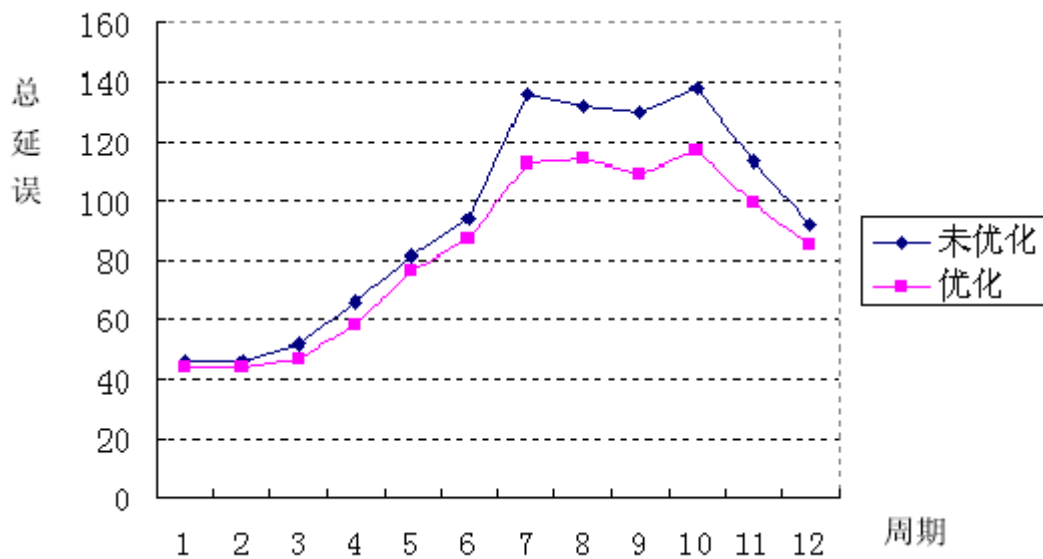


图 5.17 车辆总延误的对比

我们所建立的模型控制下的车辆总延误在非交通高峰期时优化效果一般,与传统的定时信号控制下的车辆总延误接近,主要是因为交通流小,车辆排队较少,信号灯的控​​制作用不是很明显;但是随着交通高峰期的临近,交通流逐渐增大,而交叉路口的最大通行能力是固定的^[26],传统的定时信号控制已经不能满足当前的交通能力,因此,车辆排队开始增加,造成了很大的车辆总延误,而本模型采用实时信号控制,动态的调整各个交叉路口的信号灯周期,实时优化信号灯的各个

相位, 以实现最小的车辆总延误, 因此, 与定时控制相比, 其优化效果逐渐增大, 且在交通高峰期时, 优化效果达到最大.

邻近交叉路口的周期长度对比, 如图 5.18 所示:

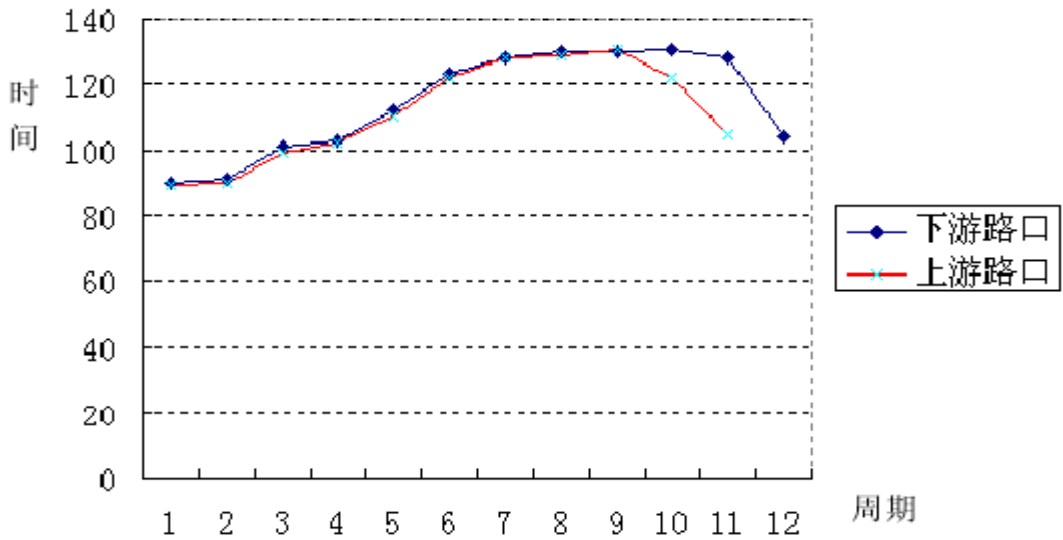


图 5.18 交叉路口周期长度的变化

与传统的定时信号控制相比, 我们的模型控制下的信号灯周期随着交通流变化而变化。在前 4 个交通状态中, 上下游路口的信号周期都随着交通流的增加而变大, 随着交通高峰期的临近, 信号灯周期开始逐渐变大, 以满足最大的车辆通过, 实现最小的车辆总延误^[27]; 当处在交通高峰期时, 信号灯周期随着交通流的稳定而逐渐稳定; 最后信号灯周期随着交通流的消退而逐渐降低, 只需要满足实际的车辆通行率, 实现最小的车辆总延误。

相位差 s 的变化, 如图 5.19 所示:

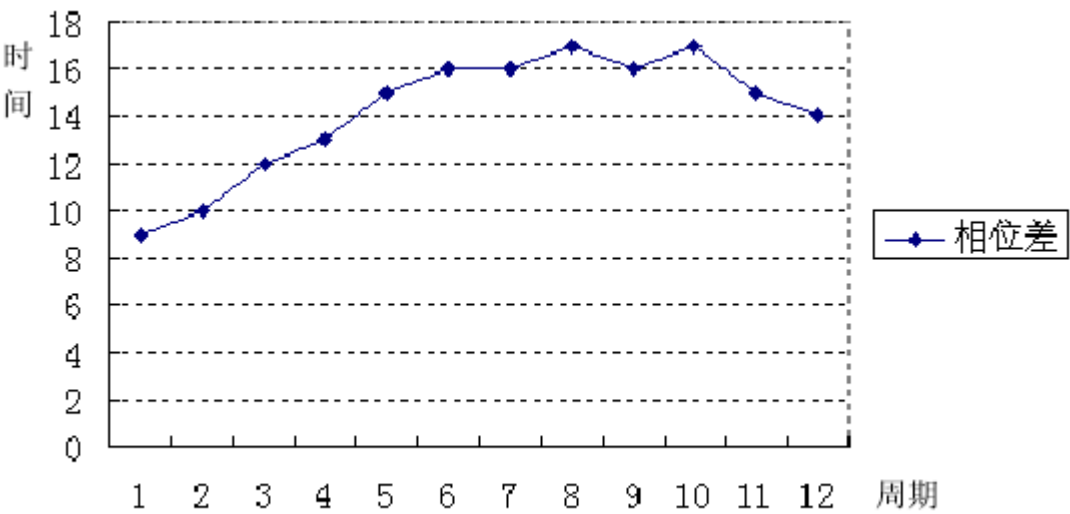


图 5.19 交叉路口相位差变化

相位差的变化随着交通流的变化而变化。根据公式 (3-30) 可知, 相位差 s

的大小主要受车速和信号周期的影响。在前四个交通状态,随着车流的逐渐增大,车速逐渐降低,信号周期逐渐变大,相位差逐渐增加。当处于交通高峰期时,车速稳定,信号周期稳定,则相位差也相对稳定。随着交通高峰期的消褪,相位差逐渐回落。

通过实验仿真,邻近交叉路口的信号灯实时控制较传统的信号灯控制方案有着很好的改进,实现了非交通高峰期时的绿波效应,大幅度的降低了交通高峰期时的车辆总延误,有利于交通高峰期时的畅通。两个交叉路口的信号周期随着交通流的变化而变化,实现了最佳的绿信比配置^[28],提高了单位信号周期的最佳服务

性能;相位差的变化主要受车速和信号周期的影响,表现为随着交通高峰期的邻近而逐渐变大,处于高峰期时处于相对稳定状态,随着交通高峰期的消褪而逐渐变小。

六、 模型的评价

6.1 优点

- 建立的交叉路口多目标动态优化配时模型具有通用型,然后建立相邻交叉路口信号灯的实时控制模型,加强信号相位控制时。采用数解法对相位差优化,加大了模型的健壮性。
- 创新性的给出了环形交叉路口的设计和分析。
- 模型做了较一般性的推广,更具普遍性和实际意义。

6.2 缺点

所建立的模型知识针对一定客流量的情况之下,在客流量达到相当大的值的时候,所建立的模型会不精确;另外,在多项交叉路口的情况下,产生一系列相位差的情况,也会造成很大的影响。

七、 模型的改进

7.1 一种环形交叉路口的创新设计

车辆按照顺时针方向单进口放行,进道口位相序按照顺时针,这样设计,车辆将会按照一个相位行驶,不会存在绿时浪费的问题,而且极大的提高了自动调控车辆的效果,但是,我们也考虑到了车流量较大的情况,针对数据分析显示,此创新的设计,适用于车流量较少的郊区地带。其它设计思路采用上面模型的设计方案,采用无交通灯控制的方案。

表 6.1 环形交叉路口相位差结果

交叉路口 理想信号编号	A	B	C	D	E	F
	①	②	③	③	④	⑤
各信号位置	右	左	左	右	左	右

绿信比 (%)	50	45	55	55	50	60
损失 (%)	15	14	2	18	5	5
有效绿信比 (%)	38	46	60	42	55	60
相位差 (%)	62.5	20.0	47, 5	47.5	20.0	57.5

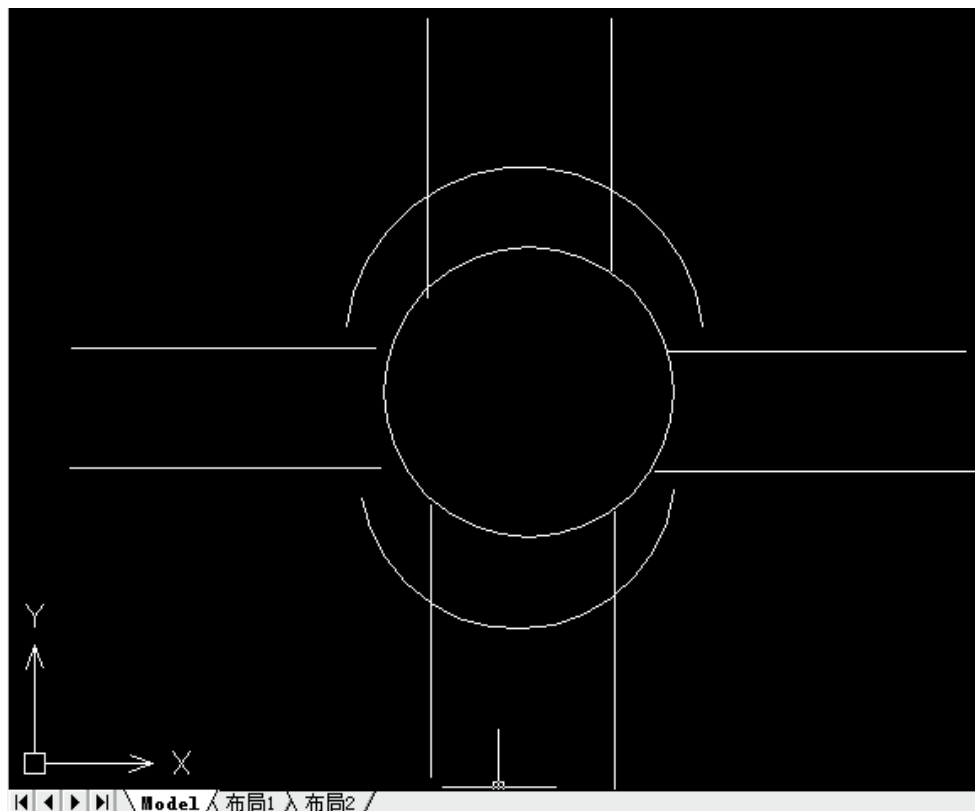


图 6.1 环形交叉路口设计简图

八、模型的推广

我们建立的模型根据数据分析和仿真实验数据检验具有很强的健壮性和可靠性, 不仅对于单交叉路口的设计具有通用性, 最重要的是适用于邻近交叉路口, 经过优化相位差, 节减少了延误时间, 设计的这种定时传统控制信号的方式, 解决了绿时浪费问题。

针对优化城市相邻两家插口信号配时问题, 我们首先从单点平面交叉口着手, 建立了基于多目标优化的单点交叉口交通灯动态配时模型, 用通行能力、延误、停车次数这三个指标通过加权作为多目标联合优化的参数指标, 把排队长度作为模糊控制指标, 并且追加有效绿灯时间, 进而得到信号配时的联合优化函数, 最终我们确定了在单点交叉口下的最佳周期 C 和各相位的绿灯有效时间。

然后, 我们在单点的交叉口的基础上进一步分析了相邻两交叉口的情况, 本文以简单绿波系统为例, 建立一个基于基本交通流模型以及交叉路口的排队分析理论, 以两个交叉路口的车辆总延误最小为目标, 以主干道的绿波效应以及中间路段的车辆排队长始终小于其最大车辆排队长度为约束条件的非线性函数模型,

并针对不同的交叉口饱和率进行信号配时优化,通过对路段和干线机动车流进行协调控制设计,在对路段和干线机动车流进行协调控制设计后,有效地减少了相交道路车流对干线车流的干扰,实现了邻近交叉路口信号灯的实时控制,保证了相交道路车流行车需求的同时,使干线车流获得最大的通行权。

再次,期望两相邻的交叉路口能更好的使协调配合工作,我们采用数解法以最大绿波带为优化目标通过寻找使得系统中各实际信号距理想信号的最大挪移量最小来对相位差进行了优化设计,最终借助题中表一数据进行了仿真检验,实现了在保证交通流量的情况下改善了车均延误时间,最后我们推广到了多个相关交叉口并借助一例进行了应用,使模型更具普遍性。

最后,我们做了从时间复杂度、空间复杂度、收敛性三个方面总结性的对比分析工作,集中展示了在应用本文模型改善后的交通信号配时方案,并进行了简单的仿真检验,可以做为指导提高实际平面交叉口的通行能力和服务水平,减少城市交通网的交通延误,改善城市交通现状的理论依据。

我们指出了所做工作的创新之处,介绍了模型的优缺点和改进方案。所以模型有很大的推广价值和应用前景。

九、参考文献

- [1]李致伟,城市单交叉路口信号控制优化方法的研究[D].北京:北京交通大学交通信息工程及控制,2008.5。
- [2]栗红强,城市交通控制信号配时参数优化方法研究[D].长春:吉林大学,2004。
- [3]冯玉英,常玉林,环形交叉口通行能力研究[J],江苏理工大学学报(自然科学版),2000.5,Vol.21(3):91—94。
- [4]项乔君,王炜,陈兵等,环形交叉路口通行能力理论模型研究[J].中国公路学报,1999.10,Vol.12(4):69—72。
- [5]全永焱,城市交通控制[M],北京:人民交通出版社,2003。
- [6]刘静、魏瑞轩、张善文,城市交叉路口交通优化模型[J],陕西师范大学学报,2003.4:10—13。
- [7]杨开春、段胜军,交通流的红绿灯模型[J],西安联合大学学报,2004.10:41—45。
- [8]王伟平,城市平面交叉口交通信号控制优化方法研究[D],青岛,山东华科技大学运筹学与控制理论,2004.5。
- [9]王玉鹏,多路口感应信号控制优化设计及其仿真[D],南京,河海大学交通运输规划与管理,2006.3。
- [10]荆晓辉,单交叉路口交通信号自适应控制系统研究与实现[D],重庆,重庆大学计算机理论与理论,2010.4。
- [11]张文峰,平面交叉口信号控制优化算法研究[D],昆明,昆明理工大学载运工具运用工程,2006.5。
- [12]董友才,双交叉路口流实时优化控制方案的研究[D],上海,上海交通大学电工理论与新技术,2006.12。

- [13]刘畅, 智能交通信号灯实时优化控制[D], 沈阳, 沈阳工业大学控制理论与控制工程, 2008.1。
- [14]高雨、叶炜、沈国江, 多相位智能交通控制器的设计[J], 江南大学学报, 2006.6,556—560。
- [15]王小平, 遗传算法——理论、应用与软件实现[M], 北京, 工业技术出版社, 1998。
- [16]李敏强、寇纪淞、林丹等, 遗传算法的基本理论与应用[M], 北京, 科学出版社, 2002。
- [17]张讲设、徐宗本、梁怡, 整体退火遗传算法及其收敛充要条件[J], 中国科学(E 辑), 1997.27(2): 154—164。
- [18]廖柏其, 概率论教程[M], 合肥, 中国科学技术大学出版社, 1998。
- [19]严士健、刘秀等, 概率论基础[M], 2 版, 北京, 科学出版社, 2009。
- [20]杨佩昆、张树升, 交通管理与控制[M], 北京, 人民交通出版社, 2003。
- [21]方珍珍, 单向交通的基本概念和实施原则[J], 才智, 2008.10:266。
- [22]张榕, 平面信号交叉口交通控制方案的优化研究[D], 北京, 北京工业大学交通运输规划与管理, 2003.5。
- [23]一峰, 单向交通的基本概念和实施原则[J], 交通运输, 2005.3:32—33。
- [24]基于微波检测器交通灯智能控制系统的设计[J], 西南民族大学学报, 自然科学版, 2007-33(4): 284—288。
- [25]查振亚, 叶向阳.智能交通灯控制系统[J].华中理工大学学报.2005, 25(3):255-260。
- [26]崔宝侠, 杨继平. 城市交通灯信号配时控制器优化的一种新策略[J].沈阳工业大学学报, 2007, 10: 554-559。
- [27]刘建峰, 城市干线交通控制方法研究[D], 兰州, 西北工业大学, 2001。
- [28]王殿海、曲大义, 一种实时动态交通量预测方法与研究[J], 中国公路学报, 1998.8,Vol.11:102—107。

十、附录

9.1 遗传算法配时优化程序源代码:

```
% This the main script implements the Nagel Schreckenberg cellular
automata
% based traffic model. Flow vs density curves are displayed.
%
% liu jiguo: 2993901967@qq.com
%
density = 0:0.02:1;
roadlength = 100;
vmax = 5;
tmax = 200;
pbrak = 0;
flux = [];
vmean = [];

for rho = density
    [R, J, V] = ns(rho, pbrak, roadlength, tmax, 0, 0);
    flux = [flux; J];
    vmean = [vmean; V];
end

% ----- density vs. velocity
% -----
figure
plot(density, vmean, 'k.', 'markersize', 15);
hold on
plot(density, min(vmax, 1./density-1), '-r', 'linewidth', 2)
ylim([0, 5.55])
legend({'Cellular automata aproach', ...
'$v(\rho) = \min\{v_{\max}, 1/\rho-1\}$', ...
'interpreter', 'latex'})
xlabel('density in vehicles/cell')
ylabel('velocity in cell/time')

% ----- density vs. flux
% -----
figure
plot(density, flux, 'k.', 'markersize', 15);
hold on;
```

```

plot(density,min(density*vmax, 1-density),'-r','linewidth',2)
legend({'Cellular automata aproach', ...
'$J(\rho) = \min\{\rho \cdot v_{\max}, 1-\rho\}$'}, ...
'interpreter','latex')
xlabel('density in vehicles/cell')
ylabel('flux in vehicles/time')

rfunction [rho, flux, vmean]=ns(rho, p, L, tmax, animation, spacetime)
%
% NS: This script implements the Nagel Schreckenberg cellular automata
% based
% traffic model. Car move forward governed by NS algorithm:
%
% 1. Acceleration. If the vehicle can speed up without hitting the speed
%    limit vmax it will add one to its velocity,  $v_n \rightarrow v_n + 1$ . Otherwise,
%    the vehicle has constant speed,  $v_n \rightarrow v_n$ .
%
% 2. Collision prevention. If the distance between the vehicle and the
%    car
%    ahead of it,  $d_n$ , is less than or equal to  $v_n$ , i.e. the  $n$ th vehicle
%    will collide if it doesn't slow down, then  $v_n \rightarrow d_n$  鉴?1.
%
% 3. Random slowing. Vehicles often slow for non-traffic reasons (cell
%    phones, coffee mugs, even laptops) and drivers occasionally make
%    irrational choices. With some probability pbrake,  $v_n \rightarrow v_n$  鉴?1,
%    presuming  $v_n > 0$ .
%
% 4. Vehicle movement. The vehicles are deterministically moved by
%    their
%    velocities,  $x_n \rightarrow x_n + v_n$  .
%
% USAGE: flux = ns(rho, p, L, tmax, isdraw)
%    rho      = density of the traffic
%    p        = probability of random braking
%    L        = length of the load
%    tmax     = number of the iterations
%    animation = if show the animation of the traffic
%    spacetime = if plot the space-time after the simulation ended.
%    flux     = flux of the traffic
%
% zhou lvwen: zhou.lv.wen@gmail.com
%
if nargin == 0;

```

```

    rho = 0.15; p = 0.25; L = 100; tmax = 5000; pchange = 0.5;
    animation = 'circle';
end

%rand('seed',1)
vmax = 5; % maximum speed
% place a distribution with density
ncar = round(L*2*rho);
rho = ncar/2/L;

xy = randperm(2*L, ncar);
[y,x] = ind2sub([2, L], xy); % y: 2 for right, 1 for left

v = vmax * ones(1, ncar); % start everyone initially at vmax

switch animation
case 'circle'; h = plotcirc(L, x, y, 2);
case 'line'; h = plotline(L, x, y, 2);
end

flux = [0 0]; % number of cars that pass through the end
vmean = [0 0];

voffset = 1;
vback = 1;
XY = [x y];
for t = 1:tmax
% determine the space vehicles have to move
    [gaps, gapfront, gapback] = gaplength(x, y, L);

% left to right & right to left
    l2r = find(y==1 & gaps>vmax+voffset & gapfront>vmax+voffset &
gapback>=vback);
    r2l = find(y==2 & gaps<vmax & gapfront>gaps &
gapback>=vback & rand(size(y))<pchange);
    y(l2r) = 2;
    y(r2l) = 1;
% acceleration
    v = min(v+1, vmax);

    gaps = gaplength(x, y, L);
%collision prevention
    v = min(v, gaps-1);

```



```

% random speed drops
    vdrops = ( rand(1,ncar)<p );
    v = max(v-vdrops,0);

% update the position
    x = x + v;
    passed = x>L;          % cars passed at time r
    x(passed) = x(passed) - L;% periodic boundary conditions

if t>tmax/2
    flux(1) = flux(1) + sum(v(y==1)/L); %flux = flux + sum(passed);
    flux(2) = flux(2) + sum(v(y==2)/L); %flux = flux + sum(passed);
    vmean(1) = vmean(1) + mean(v(y==1));
    vmean(2) = vmean(2) + mean(v(y==2));
end

switch animation
case 'circle'; plotcirc(L,x,y,0.1,h);
case 'line';   plotline(L,x,y,0.1,h);
end

end

flux = flux/(tmax/2);
vmean = vmean/(tmax/2);

%
-----
-----

```

```

function [gap, gapfront, gapback] = gaplength(x,y,L)
%
% GAPLENGTH: determine the gaps between vehicles
%
ncar = length(x);
gap = inf*ones(1, ncar);
gapfront = inf*ones(1, ncar);
gapback = inf*ones(1, ncar);
index = 1:ncar;
for i = index
    j1 = index(index~i & y==y(i));
    if ~isempty(j1)
        d1 = x(j1) - x(i);
        d1(d1<=-L/2) = d1(d1<=-L/2) + L;
    end
end

```

```

if any(d1>0)
    gap(i) = min(d1(d1>0));
end
end

    j2 = index(index~=i & y~=y(i));
if ~isempty(j2)
    d2 = x(j2) - x(i);
    d2(d2<-L/2) = d2(d2<-L/2) + L;
if any(d2>=0)
    gapfront(i) = min(d2(d2>=0));
end
    d3 = x(i) - x(j2);
    d3(d3<-L/2) = d3(d3<-L/2) + L;
if any(d3>=0)
    gapback(i) = min(d3(d3>=0));
end
end
end
end

```

```

%
-----
-----

```

```

function h = plotcirc(L, x, y, dt, h)
W = 0.05;
ncar = length(x);

theta = [(0-pi/L) : 2*pi/L : (2*pi+pi/L)];
R = ones(size(theta));
theta = [ theta; theta];
R = [R; R+W];
xc = cos(theta);    yc = sin(theta);
xinner = (R-W/2).*xc; yinner = (R-W/2).*yc;
xouter = (R+W/2).*xc; youter = (R+W/2).*yc;
i = sub2ind(size(R), y, x);
if nargin == 4
    color = randperm(ncar);
    xi = [xinner(i); xinner(i+2); xouter(i+2); xouter(i)];
    yi = [yinner(i); yinner(i+2); youter(i+2); youter(i)];
    h = fill(xi, yi, color); hold on
    plot(xinner(1,:), yinner(1,:), 'k', ...
        xouter(1,:), youter(1,:), 'k', ...
        xouter(2,:), youter(2,:), 'k', 'linewidth', 1.5);

```

```

        plot([xinner; xouter], [yinner; youter], 'k', 'linewidth', 1.5);
        axis image;
    else
        xi = [xinner(i); xinner(i+2); xouter(i+2); xouter(i)];
        yi = [yinner(i); yinner(i+2); youter(i+2); youter(i)];
        for i=1:ncar; set(h(i), 'xdata', xi(:, i), 'ydata', yi(:, i)); end
    end
    pause(dt)

```

```

%
-----
-----

```

```

function h = plotline(L, x, y, dt, h)

```

```

W = 2;
rmin = 1;
dw = 0.05;
rmax = rmin + dw*(W-1);
ncar = length(x);
ti = 0 : 2*pi/L : 2*pi;
ri = rmin:dw:rmax;

```

```

[theta, R] = meshgrid(ti, ri);

```

```

xmin = (R-dw/2).*cos(theta);
ymin = (R-dw/2).*sin(theta);
xmax = (R+dw/2).*cos(theta);
ymax = (R+dw/2).*sin(theta);

```

```

i = sub2ind(size(R), y, x);
xi = [xmin(i); xmin(i+W); xmax(i+W); xmax(i)];
yi = [ymin(i); ymin(i+W); ymax(i+W); ymax(i)];

```

```

if nargin == 6
    color = randperm(ncar);
    h = fill(xi, yi, color); hold on
    plot([xmin; xmax]', [ymin; ymax]', 'k', 'linewidth', 1.5);
    plot([xmin; xmax] , [ymin; ymax] , 'k', 'linewidth', 1.5);
    axis image;

```

```

else
    for i=1:ncar; set(h(i), 'xdata', xi(:, i), 'ydata', yi(:, i)); end
    end
    pause(dt)

```