

# 基于深度强化学习的交通灯区域 控制方法研究

作者姓名 \_\_\_\_\_丁献硕\_\_\_\_\_

指导教师姓名、职称 \_\_\_\_\_方敏 教授\_\_\_\_\_

申请学位类别 \_\_\_\_\_工学硕士\_\_\_\_\_



学校代码 10701  
分 类 号 TP39

学 号 17031211418  
密 级 公开

# 西安电子科技大学

## 硕士学位论文

### 基于深度强化学习的交通灯区域 控制方法研究

作者姓名：丁献硕

一级学科：计算机科学与技术

二级学科（研究方向）：计算机科学与技术

学位类别：工学硕士

指导教师姓名、职称：方敏 教授

学 院：计算机科学与技术学院

提交日期：2020 年 6 月



# **Research on traffic light area control method based on deep reinforcement learning**

A thesis submitted to  
XIDIAN UNIVERSITY  
in partial fulfillment of the requirements  
for the degree of Master  
in Computer Science and Technology

By  
Ding Xian Shuo  
Supervisor: Fang Min Title: Professor  
June 2020



**西安电子科技大学**  
**学位论文独创性（或创新性）声明**

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名： 丁献硕                      日 期： 2020.6.15

**西安电子科技大学**  
**关于论文使用授权的说明**

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，结合学位论文研究成果完成的论文、发明专利等成果，署名为西安电子科技大学。

保密的学位论文在\_\_\_\_年解密后适用本授权书。

本人签名： 丁献硕                      导师签名： 李敏

日 期： 2020.6.15                      日 期： 2020.6.15





## 摘要

近年来,随着中国现代化建设的稳步前进,城市化进程的不断加快,机动车的数量也随之大幅增加,交通拥堵已经成为各大城市普遍存在的问题。随着互联网科技的发展,人工智能、大数据、深度强化学习等技术成为研究的热点,相关技术被广泛应用于交通灯控制领域。为解决交通拥堵问题,寻找更加合理的交通灯控制算法提供了新的技术支持。

由于交通流特性复杂,路网规模庞大,因此将路网划分为若干个独立的子区域进行控制是必然的结果。准确的交通预测模型能够对接下来的交通状况更好的进行预判,对于结合当下和未来的交通情况,制定合理的交通灯控制算法具有重要意义。本文针对现有的交通灯控制算法存在的不足之处进行优化,结合子区划分和流量预测,提出了一种基于深度强化学习的交通灯区域控制方法。主要研究内容如下:

首先,对子区划分在交通信号控制中的作用进行了探索和研究。针对传统子区划分算法仅考虑道路静态属性的问题,对传统 Newman 快速算法进行改进。通过综合考虑交通流量、路段长度、交通流密度、排队长度、信号周期,计算交叉口之间的关联程度,实现了对路网的动态划分。通过实验证明基于改进 Newman 快速算法的动态控制子区划分方法优于传统 Newman 快速算法,可以实现路网子区动态划分,且结果更加符合交通流特征。

其次,针对现有相关研究中基于神经网络的交通流预测方法,没有充分考虑交通流的空间特征以及计算复杂度较高的问题,通过将 CNN 网络与 GRU 网络相结合,建立了一种新的交通流预测模型。首先使用 CNN 网络提取交通流的空间特征,然后使用 GRU 网络提取交通流的多种时间特征,最后对预测结果进行融合。实验表明,基于 CNN 和 GRU 网络的交通流预测算法优于传统 LSTM 预测算法,能够提高预测准确率。

最后,针对现有的信号控制算法没有很好解决多交叉口协同控制,且控制策略具有一定的滞后性的问题。提出了一种结合流量预测的 Double DQN 交通灯区域控制方法。首先使用子区划分算法决定交通灯协同控制的区域范围,然后利用流量预测算法提高控制策略的前瞻性,最后通过基于 Double DQN 的交通灯区域控制方法实现了对多交叉口的协同控制,仿真实验证明,在轻度交通流和重度交通流环境下,算法均能降低车辆的等待时间。提高路网的通行效率,保障路网的畅通。

**关键词:** 交通子区划分, 交通流量预测, 深度强化学习, 交通信号协同控制



## ABSTRACT

In recent years, with the steady progress of China's modernization drive, the urbanization process has been accelerating, the number of motor vehicles has increased significantly, and traffic congestion has become a widespread problem in major cities. With the development of Internet technology, artificial intelligence, big data, deep reinforcement learning and other technologies have become the research hotspot, related technologies are widely used in the field of traffic light control. In order to solve this problem, we need to find a more reasonable algorithm of traffic light control. The search for more reasonable traffic light control algorithms provides new technical support for solving traffic congestion problems.

Because of the complicated traffic flow characteristics and the large size of the road network, it is inevitable to divide the road network into several independent sub-regions for control. The accurate traffic prediction model can better predict the next traffic conditions, which is of great significance for combining the current and future traffic conditions to develop a reasonable traffic light control algorithm. This paper optimizes the shortcomings of the existing traffic light control algorithm. Combined with sub zone division and flow prediction, a traffic light area control method based on deep reinforcement learning is proposed. The main research contents are as follows:

Firstly, the role of sub-division in traffic signal control is explored and studied. Aiming at the problem that the traditional subdivision algorithm only considers the static attributes of the road, the traditional Newman fast algorithm is improved by comprehensively considering traffic flow, road length, traffic flow density, queuing length and signal period, the correlation degree between intersections is calculated. The dynamic division of road network is realized. The experimental results show that the dynamic control subarea division method based on the improved Newman fast algorithm is better than the traditional Newman fast algorithm, and can achieve the dynamic subarea division, and the results are more consistent with the characteristics of traffic flow.

Secondly, the traffic flow prediction method based on neural network in the existing related research does not consider the spatial characteristics of the traffic flow and the

problem of high computational complexity. By combining the CNN network with the GRU network, a new traffic flow prediction model is established. First, the CNN network is used to extract the spatial features of the traffic flow, then the GRU network is used to extract various temporal features of the traffic flow, and finally the prediction results are fused. Experiments show that the traffic flow prediction algorithm based on CNN and GRU network is superior to the traditional LSTM prediction algorithm and can improve the prediction accuracy.

Finally, the existing signal control algorithm does not solve the problem of multi-intersection cooperative control, and the control strategy has a certain lag. This paper presents a double DQN traffic light area control method combined with flow prediction. The traffic zone division algorithm is used to determine the area range of traffic light collaborative control, and the traffic flow prediction algorithm is used to improve the forward-looking control strategy. Based on double DQN algorithm, the collaborative control of multiple intersections is realized. Simulation experiments show that the algorithm can reduce the waiting time of vehicles in light traffic flow and heavy traffic flow environment. Improve the traffic efficiency of the road network and ensure the smooth flow of the road network.

**Keywords:** traffic zone division, traffic flow prediction, deep reinforcement learning, cooperative traffic signal control

## 插图索引

图 2.1	各因素对相邻交叉口的关联程度的影响.....	8
图 2.2	卷积神经网络结构图.....	11
图 2.3	卷积神经网络的计算流程图.....	12
图 2.4	GRU 神经网络结构.....	13
图 2.5	GRU 神经网络中的一个神经元.....	13
图 3.1	网络中的社团结构.....	17
图 3.2	仿真的交通路网.....	23
图 3.3	路网的简化图.....	24
图 3.4	传统 Newman 快速算法交通控制子区.....	24
图 3.5	传统 Newman 快速算法子区划分结果.....	25
图 3.6	传统 Newman 快速算法子区划分模块度.....	25
图 3.7	改进后的 Newman 快速算法交通控制子区组成.....	26
图 3.8	改进后的 Newman 快速算法子区划分结果.....	27
图 3.9	改进后的 Newman 快速算法子区划分模块度.....	27
图 3.10	改进后的 Newman 快速算法交通控制子区组成（第二次）.....	28
图 3.11	改进后的 Newman 快速算法子区划分结果（第二次）.....	29
图 3.12	改进后的 Newman 快速算法子区划分模块度（第二次）.....	29
图 4.1	交通流预测模型流程图.....	32
图 4.2	模型的基本框架.....	33
图 4.3	CNN 网络结构图.....	34
图 4.4	某一路段连续 3 天的交通流量.....	35
图 4.5	CNN+GRU 预测值与真实值对比.....	37
图 4.6	LSTM 预测值与真实值对比.....	37
图 5.1	路网中的一个交叉口.....	40
图 5.2	DQN 算法框架.....	42
图 5.3	结合流量预测的 Double DQN 算法框架.....	43
图 5.4	道路状态.....	45
图 5.5	轻度交通流环境下 5 种算法的 ATWT 图像.....	49
图 5.6	轻度交通流环境下 5 种算法的 AJWT 的图像.....	50
图 5.7	重度交通流环境下 5 种方法的 ATWT 图像.....	50
图 5.8	重度交通流环境下 5 种方法的 AJWT 的图像.....	51



## 表格索引

表 3.1	相邻交叉口关联程度.....	25
表 3.2	相邻交叉口关联程度（第二次） .....	28
表 4.1	交通流量时间序列.....	36
表 4.2	不同模型的预测误差.....	37
表 5.1	交叉口相位与动作对应关系.....	40
表 5.2	实际动作空间.....	45
表 5.3	TC-TFP 算法参数设置.....	48
表 5.4	轻度交通流环境下各种算法 5 次仿真试验平均评价指标.....	49
表 5.5	重度交通流环境下各种算法 5 次仿真试验平均评价指标.....	51





## 符号对照表

符号	符号名称
$i$	路网中任意交叉口
$Agent_i$	交叉口 $i$ 的信号控制单元
$A_i$	路网中交叉口信号控制单元 $Agent_i$ 的相位集合
$a_i$	路网中交叉口信号控制单元 $Agent_i$ 在某个时刻的相位动作
$L$	路网中全部车道的集合
$l_i$	交叉口 $i$ 的其中一个驶入车道
$TL$	路网中所有信号灯的集合
$tl_i$	驶入车道 $l_i$ 上的信号灯
$tla$	路网中所有信号灯 $tl$ 的动作集合
$s_{pl_i}$	车辆的当前位置, 终点, 所在车道上的信号灯
$id$	交通灯相位编号
$gain$	车辆收益
$R$	即时奖励
$\gamma$	折扣因子
$\alpha$	学习率
$\theta$	评估网络的参数
$\theta^-$	目标网络参数
$\varepsilon$	探索率
$V^i$	路网中第 $i$ 个交叉口的偏好向量
$r_{ij}$	交叉口 $i$ 和 $j$ 的关联程度
$L$	相邻交叉口间距
$C$	信号周期时长
$P$	道路密度
$X$	路段的交通流量序列
$x$	交通流量
$minX$	交通流量中的最小值
$maxX$	交通流量中的最大值
$y_i$	真实流量值
$\hat{y}_i$	算法输出的预测流量值



## 缩略语对照表

缩略语	英文全称	中文对照
ATWT	Average Trip Waiting Time	车辆平均行驶等待时间
AJWT	Average Junction Waiting Time	交叉口平均等待时间
BPN	Back Propagation neural network	反向传播神经网络
CNN	Convolutional Neural Network	卷积神经网络
DDQN	Double Deep Q-Network	双 Q 学习网络
DL	Deep Learning	深度学习
DNN	Deep Neural Networks	深度神经网络
Dueling DQN	Double Deep Q-Network	对抗深度 Q 学习网络
Distributional DQN	Distributional Deep Q-Network	分布式深度 Q 学习网络
DQN	Deep Q-Network	深度 Q 学习网络
DRL	Deep Reinforcement Learning	深度强化学习
GLD	green light district	绿灯区域仿真
GRU	Gated Recurrent Unit	门控循环单元
ITS	Intelligent Transportation System	智能交通系统
LSTM	Long Short-Term Memory	长短期记忆神经网络
MAE	Mean Absolute Error	平均绝对误差
MLP	Multi-layer perceptron neural networks	多层感知器神经网络
RE	Relative Error	相对误差
RMSE	Root Mean Squared Error	均方根误差
RL	Reinforcement Learning	强化学习
TVA	Total Vehicles Arrived	已经到达终点的车辆总数
TCL	Traffic Light Control	交通灯控制



# 目录

摘要.....	I
ABSTRACT.....	III
插图索引.....	V
表格索引.....	VII
符号对照表.....	IX
缩略语对照表.....	XI
第一章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.3 本文主要工作和组织结构.....	4
第二章 相关理论与技术.....	7
2.1 区域路网子区划分.....	7
2.1.1 路网子区划分的目的和意义.....	7
2.1.2 影响路网子区划分因素.....	7
2.2 预测模型介绍.....	10
2.2.1 交通流量预测模型介绍.....	10
2.2.2 CNN 神经网络介绍.....	10
2.2.3 GRU 神经网络介绍.....	12
2.3 区域交通协同控制策略.....	14
2.3.1 协调控制.....	14
2.3.2 分散控制.....	15
2.3.3 其他控制策略.....	15
2.4 本章小结.....	16
第三章 基于改进 Newman 快速算法的动态控制子区划分.....	17
3.1 社团结构定义.....	17
3.2 常见的社团划分方法.....	18
3.2.1 Kernighan-Lin 算法.....	18
3.2.2 谱平分算法.....	19
3.2.3 GN 算法.....	19
3.2.4 Newman 快速算法.....	20
3.3 基于交叉口关联程度改进的 Newman 快速算法.....	20

3.3.1 相邻交叉口关联程度计算.....	20
3.3.2 改进的 Newman 快速算法.....	21
3.4 基于改进 Newman 快速算法的动态控制子区划分建模.....	22
3.5 实验设计及结果分析.....	23
3.5.1 实验设计.....	23
3.5.2 结果分析.....	24
3.6 本章小结.....	30
<b>第四章 基于 CNN 和 GRU 网络的交通流预测方法.....</b>	<b>31</b>
4.1 交通流预测问题描述.....	31
4.1.1 交通流预测模型.....	31
4.1.2 交通流预测的误差评估.....	32
4.2 基于 CNN 和 GRU 网络的交通流预测方法建模.....	33
4.2.1 使用 CNN 网络对交通流空间特征进行提取及预测.....	33
4.2.2 使用 GRU 网络对交通流时间特征进行提取及预测.....	34
4.2.3 预测结果整合.....	35
4.2.4 参数选择.....	35
4.3 实验设计及结果分析.....	36
4.3.1 实验设计.....	36
4.3.2 结果分析.....	37
4.4 本章小结.....	38
<b>第五章 结合流量预测的 Double DQN 交通灯区域控制方法.....</b>	<b>39</b>
5.1 交通灯控制问题描述.....	39
5.1.1 交叉口通用数据描述.....	39
5.1.2 交叉口控制算法性能评价指标.....	41
5.2 结合流量预测的 Double DQN 算法.....	41
5.3 结合流量预测的 Double DQN 交通灯区域控制方法建模.....	44
5.3.1 路网交通状态空间.....	44
5.3.2 路网交叉口动作集.....	45
5.3.3 路网交叉口奖励值.....	46
5.3.4 结合流量预测的 Double DQN 交通灯区域控制算法.....	46
5.4 实验设计及结果分析.....	47
5.4.1 实验设计.....	47
5.4.2 结果分析.....	48
5.5 本章小结.....	52

第六章 总结与展望.....	53
6.1 工作总结.....	53
6.2 研究展望.....	53
参考文献.....	55
致谢.....	59
作者简介.....	61





## 第一章 绪论

### 1.1 研究背景及意义

近年来,随着中国现代化建设的稳步前进,城市化进程的不断加快,机动车的数量大幅增加,交通问题深深困扰着各大城市。城市的交通状况已经成为衡量一个城市先进性的准则,对于城市的经济发展有着重要的作用。

城市交通拥堵问题由来已久。60年代时,在一些经济发展较快的西方国家,部分大中型城市的交通问题已经日渐突出,严重制约了城市的发展。在中国,随着现代化建设的不断推进,社会和城市经济建设发展迅猛,机动车数量增长迅速,城市的交通问题也日益严重。

交通问题对人们的日常生活有着巨大影响,在世界各国都造成了巨大的经济损失。美国得克萨斯运输研究所称,美国每年因为交通问题损失超过1600亿美元。英国每年仅伦敦就因为交通问题损失超过1500万美元。日本每年因为交通问题受到超过12万亿日元的经济损失。与此同时,中国每年因交通拥堵损失约2500亿人民币。在交通拥堵的时候,由于汽油不能完全燃烧,汽车废气中将含有更多的有毒有害气体,严重影响环境卫生。此外,据统计全球每年有超过135万人死于交通事故,在五岁至二十九岁的人群中,排名第一的死亡原因就是交通事故。

为了适应车辆的快速增长,缓解日益严峻的交通问题,既可以采取修建更多的道路、扩大路网规模的方法,也可以采取“行政治堵”的方法,如单双号限行政策。但是,长期的研究和实践证明,这些方法对于解决交通问题全都作用有限。因此,只有在扩大路网规模的同时,不断提高道路基础设施的利用率,采用更高效的方法对城市路网进行管理,才能更好地解决日益严峻的交通问题。

从19世纪开始,人们就开始尝试通过各种方法解决日益严重的交通问题,针对交通流的非线性、时间相关性和不确定性等特点,需要建立全新的交通信号控制模型。日常生活中常见的交通信号灯都是采用定时信号控制方法,通过预先设置的控制方案,让信号灯以固定的周期进行轮换。这种定时信号控制方法由于不能根据实际交通情况灵活调整配时方案,因此控制效果很差。长期的研究和实践证明,把车辆和道路相结合,引入各种高新技术,是解决交通问题的最好方法。智能交通系统(Intelligent Transportation System, ITS)利用各种高新技术实现对各个交叉口的交通信号配时方案进行即时调整,大大增强了整个交通系统的通行效率。

治理交通问题最基本、最有效的方式,就是根据交通状况,合理配置控制信号的周期时长、绿信比、相位差等参数,对交通信号进行控制。但是,随着现代城市路网

规模的急剧扩大,将交通路网看成一个整体或一个个孤立个体进行信号控制都是不合理的。因此,本文主要研究对路网进行子区划分,然后构建子区内的流量预测模型,基于预测结果对子区间的多交叉口进行交通灯协同控制,用以提高交通路网的通行效率,进一步缓解道路拥挤、交通事故、环境污染等交通问题。

## 1.2 国内外研究现状

基于本文的主要研究内容,分别对交通控制子区划分方法、交通流预测方法、交通灯区域协同控制方法等在国内外的研究现状进行回顾和总结。

### (1) 交通控制子区划分方法研究

依据路网的静态空间特征和交通流的动态特征,将一个复杂且庞大的交通路网划分为若干个合理的路网子区,对于制定各个路网子区内和子区间的优化控制策略十分关键。

国外的学者很早就对交通控制子区的划分进行了研究。“交通控制子区”的概念最早由 Walinchus<sup>[1]</sup>在 1971 年提出,通过考虑路网的物理特征、交叉口饱和度、信号灯的相位差等划分路网子区。为以后研究交通控制子区的划分打下基础。Yagoda N H<sup>[2]</sup>等首次提出“关联性”的概念,它依据交叉口的交通流量与交叉口之间的距离的比值来评价相邻交叉口间的关联程度,判断两个交叉口是否需要被划分到同一个路网子区中。Pinnell C<sup>[3]</sup>等依据交叉口之间的距离、交通流量和车流的规律来计算关联程度,作为路网子区划分的依据。Moore J E<sup>[4]</sup>等对交叉口流量比进行聚类分析,根据聚类结果划分路网子区。Chang E C P<sup>[5]</sup>提出上游交叉口的控制策略将影响下游交叉口的关联度大小,并建立了相邻交叉口的关联度模型。

我国的学者们对于交通控制子区的划分也做了很多研究。李瑞敏<sup>[6]</sup>研究了如何依据交通流离散性、交通流量、主干道交通构成和信号周期,动态划分交通控制子区,提出了一种基于模糊理论的动态划分方法。徐建闽<sup>[7]</sup>等依据交叉口的饱和度计算交叉口的关联程度,作为路网子区划分的依据。Hu<sup>[8]</sup>等依据车辆的交通状态计算相邻交叉口的关联程度,对路网进行子区划分。

### (2) 交通流预测方法研究

交通流的不确定性、非线性和时间相关性等特点<sup>[9]</sup>,使交通流变得难以预测。20 世纪 90 年代开始,伴随着机器学习和深度学习理论的迅速发展,基于机器学习的交通流预测模型逐渐成为人们研究的重点。

1997 年,Smith B L<sup>[10]</sup>等人使用多种预测模型对交通流进行预测,发现非参数回归模型明显优于其他模型。Messer C<sup>[11]</sup>等提出基于径向基神经网络的短期交通流预测模型,获得了优于使用反向传播神经网络模型(Back Propagation neural network, BPN)

的实验结果。Kranti K<sup>[12]</sup>等研究发现,将交通流时间间隔设置为 5-15 分钟,对实验结果将产生积极影响。Lv Y<sup>[13]</sup>首次提出了使用深度堆叠自动编码器 SAE 方法进行交通流量预测,并使用贪婪策略进行分层训练。Ma X<sup>[14]</sup>提出将 LSTM 神经网络用于交通预测,取得了于 ARIMA、SVM、卡尔曼滤波模型、延时神经网络、时间延迟神经网络和非线性自回归神经网络的结果。随后, Ma X<sup>[15]</sup>提出了一种基于 CNN 的方法来预测大规模网络内的交通速度。Polson N G<sup>[16]</sup>提出了一种结合一系列 tanh 层和一个线性层的深度学习方法来捕获故障或拥塞对交通流预测的影响。Zheng J<sup>[17]</sup>等提出了一种基于卷积神经网络的模型,可以同时在全市范围的交通流进行预测,为预测算法提供了新思路。随后, Zheng J<sup>[18]</sup>等人又对现有模型做出改进,通过加入残差单元,提出了 ST-Res Net 模型,取得了较好的实验效果。Yuankai Wu<sup>[19]</sup>提出了一种基于深度神经网络的交通流量预测模型,该模型利用交通流量的时间周期的特征来提高预测精度。邓烜堃<sup>[20]</sup>提出了一种将卷积神经网络和长短时记忆神经网络相融合的交通流量预测方法,利用卷积神经网络提取空间特征。由于使用路网图片作为输入,所以可能引入很多不必要、不符合现实的特征,对预测结果造成不利影响。罗向龙<sup>[21]</sup>等提出将 DBN 模型与 SVR 模型相结合的交通流量预测模型,提升了预测效果。

### (3) 交通灯区域协同控制方法研究

对单个交叉口进行控制,其控制范围和对路网区域的影响都较小。随着路网规模的扩大,仅对单个交叉口进行控制已经不能满足实际需要,将多个交叉口的交通灯联合起来进行协同控制,逐渐成为人们研究的重点。

国外的学者们对于各种交通灯区域协同控制方法进行了深入研究。Kuyer L<sup>[22][23]</sup>提出采用协作图,利用邻居交叉口间的通信进行信息交换,完成多交叉口交通信号灯的相位动作选择。Li L<sup>[24]</sup>提出使用深度学习网络来解决基于强化学习的信号控制算法中存在的状态空间爆炸问题。Rosyadi A R<sup>[25]</sup>提出采用协同 Q-Learning 算法,根据邻居交叉口信号灯的收益值完成多交叉口交通信号灯的相位动作选择,实现多交叉口交通信号灯协同控制。Vidhate D A<sup>[26]</sup>提出根据等待车辆和新到达车辆的数量,结合等待时间完成相位动作的选择。Surya S<sup>[27]</sup>提出根据预测的交通流状态,计算道路的拥堵因子,获得最佳的信号持续时间。

我国的学者们对于交通灯区域协同控制方法的研究也做了许多贡献。伦立宝<sup>[28]</sup>提出采用协作图,利用启发式强化学习算法对多个交叉口的交通灯进行协同控制。夏新海<sup>[29]</sup>根据多智能体间的关系,提出了基于强化学习算法的单交叉口和多交叉口控制模型,利用分布式 Nash-Q 算法、策略梯度上升算法等对交通灯控制算法进行了改进,王莹多<sup>[30]</sup>提出基于深度强化学习算法(DQN)的单交叉口控制算法,同时提出了基于协作图和多智能体关系的迁移学习算法,取得了较好的效果。闫呈祥<sup>[31]</sup>提出一种基于分布式 W-Learning 实现交通灯分布式多路口多策略协同控制算法,实现了路口之

间的数据共享,加快了算法的学习速度。Lin<sup>[32]</sup>提出一种基于数据驱动信念规则库的智能交通灯控制方法,根据预测的交通流量和道路状况确定交通信号灯控制策略。

You-Ren Chen<sup>[33]</sup>提出将预测与交通灯控制相结合,提出了 MTLCS 模型,能够快速响应实时交通状况,减少交通拥堵的发生。

### 1.3 本文主要工作和组织结构

为了解决交通问题,需要使用合适的交通信号控制方法,本文针对现有算法存在的问题进行了以下研究:

(1) 针对传统子区划分算法仅考虑道路静态属性的问题,对 Newman 快速算法进行改进,构建边权模型,实现交通区域的子网划分。通过综合考虑交通流量、交叉口之间的距离、排队长度、信号周期、路段交通流密度,定量分析交叉口间关联度并构建边权。由于交通流是一个动态值,因此可以实现对路网的动态划分。并通过仿真软件 GLD 中进行的仿真实验表明,基于改进的 Newman 快速算法的动态控制子区划分方法优于传统 Newman 快速算法,能够敏锐的辨别路段间的耦合关系,实现区域路网合理、正确的划分,且结果更加符合交通流特征。

(2) 针对现有交通流预测方法,没有充分考虑到交通流的空间特征以及计算复杂度较高的问题。提出了一种基于 CNN 和 GRU 网络的交通流预测模型。分别使用 CNN 和 GRU 神经网络对交通流的空间特征和时间特征进行提取,提高了对交通流的预测能力。

(3) 针对现有的信号控制算法没有很好的解决多交叉口协同控制,且控制策略具有一定的滞后性的问题。通过结合子区划分和流量预测,提出一种基于流量预测的交通灯区域控制方法,实现了多交叉口协同控制,降低了车辆的等待时间,提高了车辆的通行效率,保障了城市交通体系的平稳运行。

本文共有六个章节,内容分别是:

第一章:绪论。阐述了论文的研究背景及意义,对国内外研究现状的总结,介绍本文的主要研究内容和安排。

第二章:介绍了本文需要用到的各种理论和技术。首先介绍了对区域路网进行子区划分的目的和意义,以及影响路网子区划分的各种因素,然后介绍了各种交通流量预测模型以及 CNN 和 GRU 算法,最后介绍了研究区域路网常用的控制策略。为接下来研究交通灯区域协同控制方法做铺垫。

第三章:首先介绍了社团结构的定义和几种常见的社团划分算法,然后给出了交叉口关联程度的计算方法,之后提出改进的 Newman 快速划分算法并应用于交通路网的划分。最后进行对比实验,验证改进后算法的性能。

第四章：首先介绍交通流预测算法的问题描述，然后针对交通流的空间特征和时间特征，提出了一种基于 CNN 和 GRU 网络的交通流预测模型。分别使用 CNN 和 GRU 神经网络对交通流的空间特征和时间特征进行提取，最后进行实验分析，验证预测模型的性能。

第五章：首先介绍了交通灯控制问题和算法性能的评价指标，以及交通灯控制建模方式。然后介绍了 Double DQN 算法，并提出了结合流量预测的 Double DQN 算法。之后在子区划分的基础上，使用结合流量预测的 Double DQN 算法进行交通灯区域控制算法建模。最后通过实验与其他交通控制方法进行对比。

第六章：总结与展望。对本文的工作内容进行总结，指出不足，并对下一步需要解决的问题进行展望。



## 第二章 相关理论与技术

### 2.1 区域路网子区划分

近年来,随着我国现代化建设的不断推进,社会和城市经济建设发展迅猛,城市路网的规模急剧扩大,道路状况的复杂性也随之增加。如果继续将城市路网当成一个整体进行信号协同控制,算法的复杂度将随着交叉口数量的增加而指数级增长,进而造成计算效率低下、状态空间爆炸等问题。因此人们开始将整个路网划分成一个个子区,以各个子区为控制对象,实现由部分到整体的协同控制<sup>[34]</sup>。

#### 2.1.1 路网子区划分的目的和意义

将路网划分成一个个子区,可以降低信号控制系统内的耦合性,提高信号控制系统的可靠性。如果将路网作为一个整体进行控制,当其中个别交叉口出现故障时,将影响整个控制系统,甚至造成整个路网的瘫痪。将路网划分成一个个子区,子区内的故障一般不会扩散至整个路网,不利影响将被限制在子区内部,且便于对信号控制系统进行检查和维护。

将路网划分成一个个子区,既可以降低信号控制系统内的复杂度,也可以提高信号控制系统的运行效率。交通路网信号控制系统使用计算机进行数据的处理和交互,但是计算机的存储能力和计算能力是有限的。交通路网信号控制系统内部的数据量将随着系统内交叉口数量的增加而指数级增长。因此,将路网划分成一个个子区,可以有效减少子区内控制的信号灯数量,从而减少系统内部的数据量,提高交通路网信号控制系统运行效率。

将路网划分成一个个子区,可以提高路网信号控制系统的有效性。将交通状态相似的交叉口划分到同一个子区,在各个子区内部采用适合当前交通状态的控制方案,在各个子区之间建立协作关系,进行微调,以此保障整个路网的通行效率。城市路网的规模庞大,每个路段都可能具有不同的交通特性,如果按照统一的策略进行控制,无疑是不符合实际的。将路网划分成一个个子区,能够依据各子区的特性制定控制方案,提高了控制策略的灵活性,且控制方式更加简便。

#### 2.1.2 影响路网子区划分因素

影响交通路网子区划分结果的因素有很多,可以划分为静态因素和动态因素两种类型。静态因素包含交叉口距离、道路的类型,车流运行方式等,对于划分子区的影响固定不变;动态因素包含信号灯相位差、绿信比、信号周期等信号配时参数以及交

通流量等，实时影响划分子区的策略。对路网进行控制子区划分就是依据交叉口的静态因素和动态因素，遵循一定的策略，将有利于协同控制的交叉口划分为同一个子区进行协同控制。

各因素对相邻交叉口的关联程度的影响作用如图 2.1 所示。

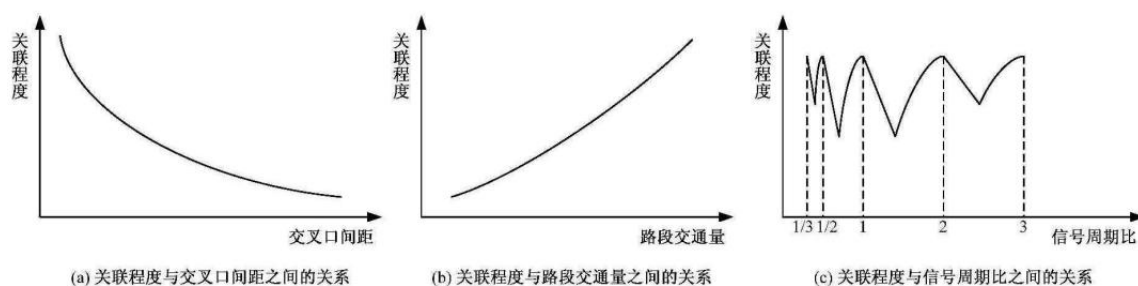


图 2.1 各因素对相邻交叉口的关联程度的影响

### 1) 交叉口距离

交叉口距离是指是连接两个交叉口的道路的长度，交叉口距离能够影响车辆的行驶时间、道路的容量等。相邻交叉口距离是影响相邻交叉口之间关联程度的最重要也是最基础的因素，两个交叉口之间的距离与交叉口之间的关联程度呈负相关，其相关关系如图 2.1 (a) 所示。如果两个相邻交叉口之间的距离较长，则下游交叉口车辆的拥堵情况不易影响到来自上游交叉口的车辆，同时，来自上游交叉口的车辆将随着路程的增长逐渐分散，车辆将随机到达下游交叉口，因此相邻交叉口之间关联程度较低，不存在协同控制的必要；如果两个相邻交叉口间距较短，则下游交叉口车辆的拥堵情况严重影响到来自上游交叉口的车辆，同时，来自上游交叉口的车辆将以车队的形式到达下游交叉口，因此相邻交叉口之间关联程度较高，存在协同控制的必要。

### 2) 交通流量

交通流量是指一段时间内，道路上经过的车辆数。交通流量的大小是影响相邻交叉口之间关联程度的重要因素之一。交通流量实时反映了路段的交通情况，动态影响相邻交叉口之间的关联程度。两个交叉口之间的交通流量情况与交叉口之间的关联程度呈正相关，其相关关系如图 2.1 (b) 所示。如果两个相邻交叉口之间的交通流量较小，交叉口之间能容纳较多车量，则下游交叉口车辆的拥堵情况不易影响到来自上游交叉口的车辆，同时，来自上游交叉口的车辆离散性较强，因此相邻交叉口之间关联程度较低，不存在协同控制的必要；如果两个相邻交叉口之间的交通流量较大，交叉口之间能容纳的车量较少，则下游交叉口车辆的拥堵情况严重影响到来自上游交叉口的车辆，同时，来自上游交叉口的车辆离散性较弱，因此相邻交叉口之间关联程度较高，存在协同控制的必要。



### 3) 排队长度

排队长度是指车辆由于遇到红灯或交通拥堵,等待时排成队列的长度。排队长度实时反映了路段的交通情况。当交通流量较大时,车辆的排队长度随信号周期进行累加,甚至堵塞上游交叉口。排队长度与相邻交叉口之间关联程度呈正相关。排队长度越大,交叉口之间关联程度越高。

### 4) 交通流密度

交通流密度是指在某一时刻,道路上车辆的数量与道路长度的比值。交通流密度实时反映了路段的交通情况,当交通流量较大时,交通流密度也较大。交通流密度与相邻交叉口之间关联程度呈正相关。交通流密度越大,交叉口之间关联程度越高。

### 5) 交叉口信号配时参数

交叉口信号配时参数主要包括信号周期、绿信比和相位差。交叉口信号配时参数是影响相邻交叉口之间关联程度的重要因素之一。

#### a 信号周期

信号周期是指红、黄、绿三种信号灯连续执行一次所需的总时间,也可以表示为连续两次亮起两次红灯之间所需的时间间隔。信号周期是影响相邻交叉口之间关联程度的重要因素之一。信号周期反映了各个交叉口的交通情况,两个交叉口之间的关联程度与两个交叉口信号周期的比值有关,其相关关系如图 2.1 (c) 所示。如果两个相邻交叉口之间的信号周期不均匀,说明交叉口之间的交通状态不相似,不容易兼顾相邻交叉口的控制效果,因此邻居交叉口之间联系不紧密,不存在协同控制的必要;如果两个相邻交叉口之间的信号周期相近,说明交叉口之间的交通状态相似,能够兼顾相邻交叉口的控制效果,因此邻居交叉口之间联系紧密,存在协同控制的必要。

#### b 绿信比

绿信比是指在一个信号周期中,绿灯时长与总时长的比值。上游交叉口与下游交叉口之间绿信比的差值,能够对相邻交叉口之间交通量产生累积影响,是影响相邻交叉口之间关联程度的因素之一。例如,当绿信比差值为负时,交叉口之间的车量将逐渐减少,因此相邻交叉口之间关联程度将逐渐降低;当绿信比差值为正时,交叉口之间的车量将逐渐增加,因此相邻交叉口之间关联程度将逐渐升高。

#### c 相位差

相位差是指两个交叉口同一相位绿灯或红灯开始时间的差值。两个交叉口之间的相位差,对一个信号周期内路段上可能存在的最大交通量产生影响,从而影响相邻交叉口之间的关联程度。需要注意的是,相位差对相邻交叉口之间的关联程度不产生累积影响,而绿信比能够对相邻交叉口之间的关联程度产生累积影响。

## 2.2 预测模型介绍

### 2.2.1 交通流量预测模型介绍

交通流量预测是指根据交通流量的实时数据和历史数据,对未来一小段时间内的交通流量进行预测。各国的学者们对此进行了大量的研究,建立了各种交通流量预测模型,以提高对交通流量的预测能力。比较著名的有神经网络模型、历史平均模型、时间序列模型和非参数回归模型等。这些预测模型大致上可以分为两类:第一类为参数回归模型、Kalman 滤波模型、时间序列模型等,第二类为基于小波分析的预测模型、多种神经网络预测模型、非参数回归模型等。第一类中的模型基于数学和物理方法,拥有明确的物理意义,第二类中的模型更关注与现实交通流的拟合,可能不具有明确的物理意义。

每种交通流量预测模型都拥有自己的优点和不足。历史平均模型的算法简单,容易实现,但是面对突发事件不能进行很好的进行预测。时间序列模型中的代表是ARIMA 自回归移动平均模型。ARIMA 模型的预测精度高,模型简单,容易实现,但是需要大量连续的历史数据,计算量较大,不能适应急剧变化的交通状况。Kalman 滤波模型<sup>[35]</sup>预测精度高,但是算法复杂,计算时间较长,不能实时进行预测。非参数回归模型的预测精度高、适用于复杂环境,但不适用于高维变量。神经网络模型的预测精度高,适用于复杂环境,但算法复杂,收敛速度慢。组合模型<sup>[36][37]</sup>是结合多个预测模型进行预测,预测精度高,但计算复杂,不合适的组合方法反而会降低预测精度。

近年来,随着机器学习的迅速发展,人们不满足于基于确定数学模型的预测方法,各种无模型的预测方法逐渐成为研究的热点,各种基于神经网络的预测方法被应用于交通流预测领域,如基于长短期记忆神经网络的预测模型,能够捕捉到交通流中的时间依赖关系,其各种改进被广泛应用于交通流预测领域<sup>[38]</sup>。同时有研究表明,采用因果预测方法,同时考虑上下游交通流量的影响,能够更好地对交通流量进行预测<sup>[39]</sup>。

### 2.2.2 CNN 神经网络介绍

1962 年,Hubel<sup>[40]</sup>等依据猫的视觉皮层的工作机理,首次提出了“感受野”的概念。1980 年,Fukushima<sup>[41]</sup>在感受野的基础上,提出了神经感知机。1998 年,Lecun Y<sup>[42]</sup>在多层感知机(Multilayer Perceptron, MLP)的基础上进行改进,提出了 Le Net-5,这是卷积神经网络(Convolutional Neural Networks, CNN)的雏形。2012 年,由于 Alex Net<sup>[43]</sup>模型在 Image Net 训练集中取得了优异成绩,卷积神经网络正式成为计算机视觉领域中的重要研究课题。随后,诸如 GoogleNet<sup>[44]</sup>、ResNet<sup>[45]</sup>等改进后的卷积神经网络模型不断地被提出,在图像、视频、音频等多种数据的处理上取得了优秀

的成果。

卷积神经网络已经成为深度学习领域的研究热点，尤其是在图像识别和模式分类方面。它利用局部感知、参数共享和池化的方法，减少了网络的权重个数和网络的复杂度，提高了计算效率，有效缓解了过拟合问题。并且卷积神经网络可以直接将图像编码作为输入，避免了对图像的预处理以及显式的特征提取。

通常的卷积神经网络拥有五层结构，包括输入层、卷积层、池化层、全连接层以及输出层。图 2.2 为卷积神经网络结构图。

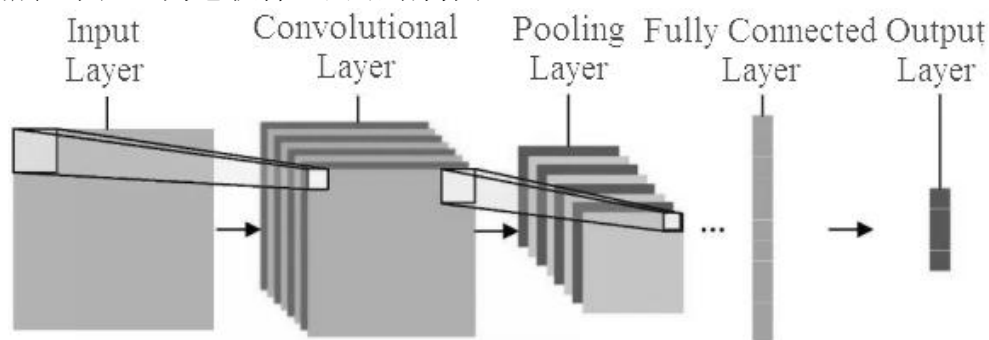


图 2.2 卷积神经网络结构图

(1) 输入层 (Input Layer)：进行数据的输入。在对输入数据进行归一化、去均值等预处理后，输入层能够实现对多维数据的处理。

(2) 卷积层 (Convolutional Layer)：卷积层负责对数据进行特征提取。每个卷积层含有多个卷积核。卷积核的大小决定了每次进行卷积时的有效范围，每次只对局部信息进行感知。在对图像进行处理时，每次使用同一个卷积核在不同的区域进行卷积，通过参数共享减少了参数的数目，降低了网络的复杂程度。

(3) 池化层 (Pooling Layer)：也叫做“下采样”、“欠采样”。池化层可以采用最大值池化、均值池化等方法，进行特征选择和信息过滤，对特征进行降维处理，减少数据和参数的数量，加速特征的提取过程，降低过拟合的风险。

(4) 全连接层 (Fully Connected Layer)：全连接层负责对卷积特征进行进一步处理，将全部局部特征组合成全局特征。并向其它全连接层进行信号的传递。

(5) 输出层 (Output Layer)：通常情况下，输出层的上游是全连接层，能够根据不同场景的输入，输出对应的数据结果。

卷积神经网络的计算流程如图 2.3 所示：

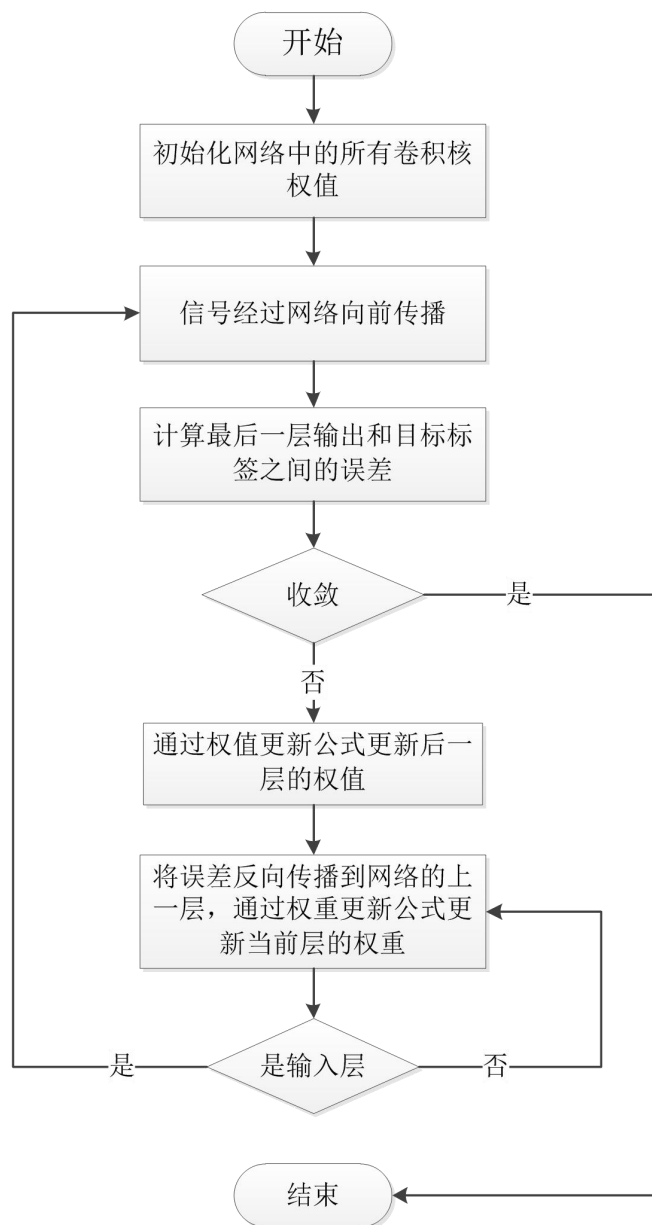


图 2.3 卷积神经网络的计算流程图

### 2.2.3 GRU 神经网络介绍

在 1997 年, Hochreiter<sup>[46]</sup>提出了长短期记忆神经网络(Long Short-Term Memory, LSTM), 这是循环神经网络(Recurrent Neural Network, RNN)模型的一种变体。在 2014 年 Cho K<sup>[47]</sup>等在长短期记忆神经网络的基础上, 改进得到了门控制循环单元神经网络(Gated Recurrent Unit, GRU), 解决了 RNN 神经网络模型的长时依赖问题, 减少了梯度爆炸或者梯度消失的发生概率。解决了长短期记忆神经网络模型结构复杂、训练时间较长的问题。长短期记忆神经网络通过遗忘门(Forget Gate)、输入门(Input Gate)和输出门(Output Gate)等门控制结构, 完成对神经细胞单元(Cell State)输入信息的消除或增强, 达到控制细胞状态的目的。而 GRU 神经网络对“门”的设计进行了

改进，通过将遗忘门和输入门合并为更新门(Update Gate)，使用重置门 (Reset Gate) 代替输出门，同时对细胞单元做了一些改进，将细胞结构由原来由三个“门”减少为两个“门”，降低了参数的数量，有效减少了训练时间。GRU 神经网络结构如图 2.4 所示。

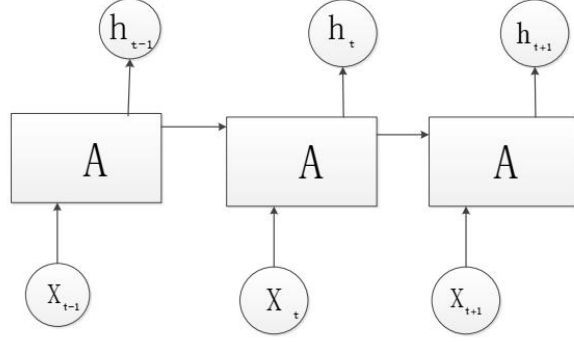


图 2.4 GRU 神经网络结构

门控制循环单元神经网络与循环神经网络有相似的地方，它们都是由一系列重复的神经单元模块组成的链式模型。不同的是，在门控制循环单元神经网络里，神经元中包含复杂的门结构，而在循环神经网络中可能只是一个 Relu、tanh 函数。循环神经网络相邻层的节点全部进行连接，但每层内部的节点不进行连接，不能对时序化的数据进行处理和建模。门控制循环单元神经网络通过在隐藏层节点之间进行连接，同时对节点的输出进行控制，能够对时序化的数据进行处理和建模。图 2.5 为 GRU 神经网络中的一个神经元。

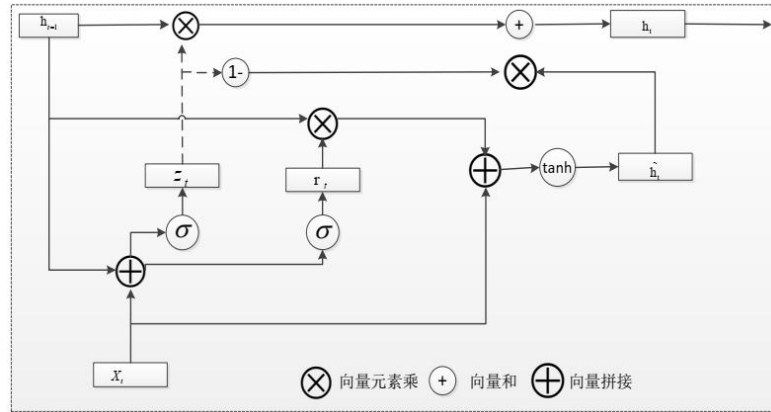


图 2.5 GRU 神经网络中的一个神经元

在 GRU 神经网络中， $\tilde{h}_t$  为候选状态：

$$\tilde{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (2-1)$$

重置门  $r_t$  决定候选状态的  $\tilde{h}_t$  对上一时刻的状态  $h_{t-1}$  的依赖程度，更新方式为：

$$r_t = \delta(W_r x_t + U_r h_{t-1} + b_r), \quad r_t \in [0, 1] \quad (2-2)$$

$r$  越小表示候选状态  $\tilde{h}_t$  对上一时刻的状态  $h_{t-1}$  的依赖程度越低,  $r$  越大表示候选状态  $\tilde{h}_t$  对上一时刻的状态  $h_{t-1}$  的依赖程度越高。

隐含状态  $h_t$  更新方式为:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \quad (2-3)$$

更新门  $z_t$  决定当前状态从上一时刻的状态  $h_{t-1}$  中保留信息的程度, 以及从候选状态  $\tilde{h}_t$  中接收信息的程度。

$$z_t = \delta(W_z x_t + U_z h_{t-1} + b_z), \quad z_t \in [0, 1] \quad (2-4)$$

$h_t$  与  $h_{t-1}$  之间存线性关系和非线性关系。当  $z_t = 0$ ,  $r_t = 1$  时, 门控制循环单元神经网络退化成简单循环神经网络。

隐状态  $h_t$ :

$$h_t = \tanh(W_h x_t + U_h h_{t-1} + b_h) \quad (2-5)$$

## 2.3 区域交通协同控制策略

由于交通路网的复杂性, 各交叉口之间存在各种直接或间接的联系。当一个交叉口的控制策略发生变化时, 路网中的交通流状态随之变化, 进而影响邻居交叉口的状态。因此, 在对交通路网进行控制时, 如果只是针对某个交叉口的控制策略进行优化, 在提高该交叉口通行效率的同时, 可能导致其它交叉口的通行效率降低, 整个路网的通行能力不增反降。只有将路网区域看成一个整体进行优化控制, 在对路网中的各交叉口控制策略进行优化时, 兼顾对整个路网的影响, 才能提高整个路网的通行效率。

随着城市路网的规模不断扩大。要想控制庞大的路网, 首先信号控制系统需要支持各个交叉口之间的信息共享, 这也是实现多子区交通信号控制策略的基础<sup>[48]</sup>。其次, 将信号控制系统设计为主从式或者分布式的多层结构, 上层的信号控制单元负责完成各个交叉口的信息交互, 底层的各交叉口信号控制单元根据得到的信息负责完成具体的控制策略, 实现区域交通信号的协同控制。

### 2.3.1 协调控制

协调控制是指将整个路网中的所有交叉口看作一个整体进行控制。由中央控制单元实现对区域内全部交叉口的统一控制, 中央控制单元能够获得整个路网的交通信息, 并制定相应的控制策略, 以此提高路网的整体通行效率<sup>[49]</sup>。近几年来, 区域协调控制已经成为交通信号控制领域的研究热点, 各种基于协调控制的算法都取得了较好的实验结果。

通常情况下, 交通信号控制系统是把一定区域的路网看作一个整体进行协同控制, 通过调整各个交叉口的控制信号, 实现对路网内交通流量的控制, 提高路网的通

行效率。路网交通信号控制系统的上层为中央控制单元，依据整个路网的交通信息对整个交通路网的控制参数进行控制，比如各个交叉口的绿信比、相位差以及公共周期等。底层为各个交叉口的信号控制单元，根据中央控制单元制定的策略，调整各个交叉口的绿信比、相位时长、信号周期等。还可以设置中间层进一步细化对路网内子区的协同控制。

现有的研究中，已经有多种基于协调控制策略的交通信号控制方法。如：依据绿灯检测、相位选择以及相位转化决策等组成控制器，调整相位时长和相位顺序；依据相邻交叉口的交通流量和交通状况进行协同控制，这种方法不需要设定公共周期；基于多智能体技术的集成交通系统的控制结构等。

### 2.3.2 分散控制

分散控制是指根据既定的控制策略，对交通网络中所有交叉口进行优化控制，提高整个路网的通行效率。在分散控制策略中，各个交叉口的信号控制单元对各自负责的交叉口进行控制，各个信号控制单元优化目标是使整个路网的通行效率最大，而不是各自交叉口的通行效率最大。在进行分散控制时，交叉口之间必须进行信息的交互，同时，由于各个交叉口之间的控制策略会互相影响，因此不能直接得到最优控制方案，必须进行逐次循环优化，最终获得使整个路网的通行效率最大的控制方案<sup>[50]</sup>。

常见的基于分散控制的控制方案中每个交叉口具有一个信号控制单元，每个信号控制单元根据当前交叉口以及邻居交叉口的交通状态，对当前交叉口的信号周期、相位时长、绿信比等信号配时参数进行控制。

### 2.3.3 其他控制策略

#### 1) 结合法

结合法将路网中的道路看成是一条条“连线”，通过对“连线”进行两两合并，逐一的对相邻交叉口之间的相位差进行协同优化，最后实现路网中所有交叉口的协同控制，达到提高路网的通行效率的目的。结合法通过调节各个交叉口之间的相位差来减少路网中车辆的等待时间，这需要较高的数据处理能力，限制了这种方法在各种区域交通控制系统中的应用<sup>[51]</sup>。

#### 2) 爬山法

爬山法也叫瞎子摸象法、逐个修改法或 K-means 方法。这种方法的中心思想是“以退为进”，针对现有控制策略中存在的问题，基于一定的条件，增加这一状态与目标之间的差异，以迂回前进的形式解决问题，达成提高路网的通行效率的目的。爬山法具有很好的性能，但是十分依赖对初始信号配时参数的设定，不合理的初始信号配时参数，将对寻找最优的信号配时参数产生严重的负面影响<sup>[51]</sup>。

### 3) 线控组合

线控组合法是通过将对整个路网的控制转变为对若干条干线进行控制,甚至转变为对两个相邻交叉口的控制。首先根据路网的各种静态因素和动态因素,预先制定多种不同的控制策略,这些策略既可以是单交叉口控制策略,也可以是多交叉口协同控制策略,然后依据路网的特性实时选择合适的控制方案。这种控制策略通过各个交叉口之间直接或间接的关系,实现对路网的区域协同控制。

## 2.4 本章小结

本章介绍了本文需要用到的各种理论和技术。首先介绍了对区域路网进行子区划分的目的和意义,以及影响路网子区划分的各种因素,然后介绍了各种交通流量预测模型以及 CNN 和 GRU 算法,最后介绍了研究区域路网常用的控制策略。为接下来研究路网区域协同控制算法做铺垫。



## 第三章 基于改进 Newman 快速算法的动态控制子区划分

城市路网规模庞大，如果将它当成一个整体进行信号协同控制，算法的复杂度将随着交叉口数量的增加而指数级增长。因此将整个路网划分成一个个子区，以各个子区为控制对象，可以降低协同控制的复杂度，提高优化效率。由于交通路网是一种复杂网络，因此可以使用社团划分算法对网络中的社团结构进行划分。本章将研究常用的社团划分算法，然后对 Newman 快速算法进行研究和改进，使其能够应用于路网的动态控制子区划分。最后仿真一段路网区域，分别使用改进前后的 Newman 快速算法进行路网控制子区划分，对划分结果进行对比和分析，验证算法的可行性与优越性。

### 3.1 社团结构定义

现实生活中存在着各式各样的网络结构，在大部分网络中，节点之间的边在网络中的分布并不均匀，某些区域的节点之间边数较多，连接密度较高，而某些区域的节点之间边数较少，连接密度较低。这种现象就是网络中的社团结构，也被称作聚类性。在一个复杂网络中，通常包含有多个社团结构，社团内部的节点之间连接紧密，边数较多，社团与社团之间的连接松散，边数较少。图 3.1 所示的网络中包含 3 个社团，每个阴影部分内的节点构成一个社团。

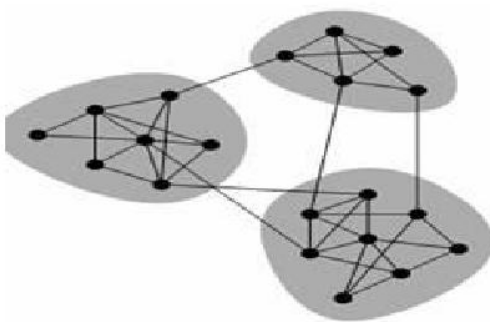


图 3.1 网络中的社团结构

目前为止，对于社团结构还没有一种公认的定义方法。一种常用的定义方法是使用相对连接频数来定义社团，就是将网络中的节点划分为小组，使小组内的节点联系紧密，小组间的联系松散。但是由于“紧密”和“松散”没有明确的判定方法，对于网络探索的可用性有限，于是研究者又提出了强社团和弱社团的概念。强社团指社团内任意一个节点的内部连接大于其外部连接，弱社团指社团内全部节点的内部连接之和大于外部连接之和。计算方法如下：

$$\xi(u_i, u_j) = \begin{cases} 1; & \text{节点} i \text{与节点} j \text{属于同一社团} \\ 0; & \text{节点} i \text{与节点} j \text{属于不同社团} \end{cases} \quad (3-1)$$

$$\varsigma(u_i, u_j) = \begin{cases} 1; & \text{节点} i \text{与节点} j \text{属于不同社团} \\ 0; & \text{节点} i \text{与节点} j \text{属于同一社团} \end{cases} \quad (3-2)$$

$$k_{in} = \sum_{u_i, u_j \in U, u_i \neq u_j} \xi(u_i, u_j) \quad (3-3)$$

$$k_{out} = \sum_{u_i, u_j \in U, u_i \neq u_j} \varsigma(u_i, u_j) \quad (3-4)$$

其中， $U$  是网络中节点的集合， $u_i$ ， $u_j$  为网络中的任意节点。 $k_{in}$  为节点的内部连接数， $k_{out}$  为节点的外部连接数。

在人际关系、计算机科学、生物科技等中都含有社团结构。如：可以依据蛋白质的功能划分蛋白质组的社团结构，因为拥有相同功能的蛋白质联系更加紧密；可以依据研究方向划分研究点的社团结构，因为相同或相似的研究方向之间有更强的索引关系；可以依据人们的交流情况划分人际关系的社团结构，因为交流较多的人之间有更强的人际关系。在交通路网中，也包含有社团结构，因此可以利用复杂网络中的社团划分方法对路网进行子区划分。

## 3.2 常见的社团划分方法

### 3.2.1 Kernighan-Lin 算法

Kernighan-Lin<sup>[52]</sup>算法是一种试探性优化算法，采用贪婪法和二分法的思想。通过将网络划分为两个社团结构，不断交换不同社团中的节点并计算增益函数  $Q$ ，将使  $Q$  值最大的交换所形成的社团结构作为最终的划分结果。增益函数  $Q$  的计算方法如公式 (3-6) 所示：

$$\delta(u_i, u_j) = \begin{cases} -1; & \text{节点} i \text{与节点} j \text{相连且属于不同社团} \\ 1; & \text{节点} i \text{与节点} j \text{相连且属于同一社团} \end{cases} \quad (3-5)$$

$$Q = \sum_{u_i \neq u_j} \delta(u_i, u_j) \quad (3-6)$$

增益函数  $Q$  表示社团内节点之间的边数与不同社团的节点之间的边数的差值。

Kernighan-Lin 算法的具体步骤：

步骤 1：将网络随机划分成两个社团，因为社团中节点的个数将被固定，所以必须保证社团大小符合现实需求。

步骤 2：选择两个社团中的任意两个节点进行交换，并计算节点交换后增益函数  $Q$  的增量  $\Delta Q$ ，将产生最大  $\Delta Q$  值的两个节点进行交换，计算并保存交换后网络的  $Q$  值。每个节点只能进行一次交换。不断迭代上述过程，直到某一个社团中的节点全部遍历完成。

步骤 3：对所有的  $Q$  值进行对比，选择最大  $Q$  值所对应的社团结构作为最终的社团划分结果。

由于 Kernighan-Lin 算法要求网络必须有且仅有两个社团结构，而且必须提前确定每一个社团中节点的个数，否则很难得到合适的划分结果。因此 Kernighan-Lin 算法很难应用于实际网络当中。

### 3.2.2 谱平分算法

谱平分算法<sup>[53]</sup>的基本理论：Laplace 矩阵中不为 0 的特征值所对应的特征向量的值，在同一个社团内的节点对应的特征向量的值是近似相等的。一个含有  $n$  个节点的无向图对应的 Laplace 矩阵  $L$  是一个  $n \times n$  维的对称矩阵。 $L$  的对角线上的元素  $l_{ii}$  是节点  $i$  的度，非对角线上的元素  $l_{ij}$  表示节点  $i$  和节点  $j$  的连接关系， $l_{ij}=-1$  表示两节点之间存在边， $l_{ij}=0$  表示两节点之间不存在边。依据网络的 Laplace 矩阵  $L$  中的次小特征值  $\lambda_2$  对社团进行划分， $\lambda_2$  值越小，表示划分结果越好。

谱平分法算法简单，鲁棒性好，当网络需要被近似地划分为两个社团时，谱平分法能够得到较好的结果。但是当网络不能被明显的划分为两个社团结构时，谱平分法的计算速度较慢。而且由于谱平分法是基于二分法的，所以网络只能被划分两个社团结构，如果需要划分为多个社团，必须对已划分的社团迭代执行该算法<sup>[54]</sup>。

### 3.2.3 GN 算法

GN<sup>[55]</sup>算法是 Girvan 和 Newman 基于分裂思想提出的一种社团划分方法。该算法不断计算所有边的边介数，删除边介数最大的边。直到网络中不存在边，网络被划分为一个个独立的小社团。边介数是指网络中经过每条边的最短路径的数目，边介数越大的边在网络中就越重要，属于某一个社团的可能性就越小，属于不同社团之间连接边的可能性越大。

GN 算法的具体步骤如下：

步骤 1：计算网络中所有边的边介数。

步骤 2：从网络中删除边介数最大的边，重新计算网络中所有边的边介数。

步骤 3：重复步骤 2，直到网络中的边全部删除，每个节点都表示一个社团。

GN 算法能够很好的对社团结构进行划分，但是，GN 算法没有定量的定义网络中的社团结构，不能直接判断划分出的社团结构是否具有现实意义。因此，还需要额外的信息，根据某种评价方式选择最终的划分结果。

GN 算法实现简单，但是因为每次删除边后都需要重新计算所有边的边介数，所以算法的时间复杂度较高。对于含有  $m$  条边  $n$  个节点的网络，最坏情况下的时间复杂度为  $O(m^2n)$ ，稀疏网络下为  $O(n^2)$ 。因此 GN 算法不适用于大规模的复杂网络。

### 3.2.4 Newman 快速算法

Newman<sup>[56][57]</sup>等基于 GN 算法提出了 Newman 快速算法，这是一种适用于大规模网络划分的凝聚算法。Newman 快速算法采用贪婪策略，在每次选择时，总是挑选能获取最大收益的方法。Newman 快速算法将网络中的每个节点当做一个单独的社团，通过模块度  $Q$  指引节点合并的方向，选择能使模块度增量  $\Delta Q$  最大的方向进行合并，当只剩一个社团时停止合并，然后选取模块度  $Q$  最大的划分结果作为最终的划分结果。

模块度  $Q$  越大则表明社团划分效果越好。若社团内部边的比例小于等于全部社团之间连接边比例的预期值，则  $Q \leq 0$ 。社团结构划分越好， $Q$  越靠近 1。通常情况下， $Q$  处于 0.3 到 0.7 之间。在划分社团时，计算各种划分方法对应的  $Q$  值，最大  $Q$  值对应的划分就是最优的划分结果。

Newman 快速算法的具体步骤如下：

步骤 1：在含有  $n$  个节点的网络中，将每个节点视为一个社团，共可划分成  $n$  个社团。模块度  $Q=0$ ，根据公式 (3-7) (3-8) 初始化矩阵  $E$  和数组  $A$  中的所有元素  $e_{ij}$ 、 $a_i$ 。

$$e_{ij} = \begin{cases} 1/(2m); & \text{节点 } i \text{ 与节点 } j \text{ 之间有边相连} \\ 0; & \text{其他} \end{cases} \quad (3-7)$$

$$a_i = k_i / (2m) \quad (3-8)$$

式中， $k_i$  为节点  $i$  的度， $m$  为网络中边的总数。

步骤 2：选择两个有边相连的社团进行合并，使用公式 (3-9) 计算合并前后模块度的增量  $\Delta Q$ 。

$$\Delta Q = e_{ij} + e_{ji} - 2a_i a_j = 2(e_{ij} - a_i a_j) \quad (3-9)$$

选择  $\Delta Q$  最大的社团进行合并，更新矩阵  $E$  和模块度  $Q$ 。

步骤 3：重复步骤 2，直至全部节点合并为 1 个社团，算法结束。

在 Newman 快速算法的运行过程中，最多执行  $n-1$  次社团合并。Newman 快速算法会输出与一个模块度  $Q$  有关的树状图，选取令模块度  $Q$  最大的划分结果作为最终的社团划分结果。

## 3.3 基于交叉口关联程度改进的 Newman 快速算法

### 3.3.1 相邻交叉口关联程度计算

在计算相邻交叉口的关联程度时，需要综合考量多个影响因素。因此，本文首先定义路网中各个交叉口的偏好向量，然后依据偏好向量计算各个交叉口的关联度。

在一个包含  $m$  个交叉口的路网中，各个交叉口考虑受  $n$  个因素的影响，路网中的第  $i$  个交叉口受到这  $n$  个因素的影响程度为  $v_j^i$ ，并且  $v_j^i \geq 0 (j=1,2,\dots,n)$ ，则称向量  $V^i = (v_1^i, v_2^i, \dots, v_n^i)$  为路网中第  $i$  个交叉口的偏好向量， $i=1,2,\dots,m$ 。

影响相邻交叉口关联程度大小的因素有许多，本文采用了其中 5 个影响因素，即相邻交叉口间距  $L$ 、信号周期时长  $C$ 、交通流量  $q$ 、排队长度  $Q$  和道路密度  $P$ 。每个交叉口可以获得各个影响因素对应的值，经过归一化后，第  $i$  个交叉口的偏好向量为  $V^i = (C^i, q^i, L^i, Q^i, P^i)$ 。相邻交叉口间距  $L$  越大，其偏好向量值越小，相邻交叉口关联程度越小，越不容易聚集；而信号周期时长  $C$  越接近、交通流量  $q$  越大、排队长度  $Q$  越大、道路密度  $P$  越大，其偏好向量值越大，相邻交叉口关联程度越大，越容易聚集。

相邻两个交叉口偏好向量  $V^i$  和  $V^j$  之间的关联程度  $r_{ij}(V^i, V^j)$  计算公式为 (3-10)：

$$r_{ij}(V^i, V^j) = \frac{(|V^i - \bar{V}^i|)(|V^j - \bar{V}^j|)^T}{\|V^i - \bar{V}^i\| \|V^j - \bar{V}^j\|} \quad (3-10)$$

其中， $\|\cdot\|$  是矢量的 2 范数， $\bar{V}^i = \frac{1}{n} \sum_{l=1}^n v_l^i$ ， $\bar{V}^j = \frac{1}{n} \sum_{l=1}^n v_l^j$ 。并且  $0 < r_{ij}(V^i, V^j) < 1$ 。

### 3.3.2 改进的 Newman 快速算法

对交通路网进行子区划分，就是要将交通流状态相似、关联程度较高的交叉口划分到同一个子区中，再以各个子区为控制对象，就可以降低协同控制的复杂度，提高优化效率。因此在 Newman 快速算法中，考虑交通流状态对于路网划分的影响，引入交叉口关联程度作为节点之间的边权参数，建立适用于路网的动态子区划分模型。 $r_{ij}$  作为道路网络中节点  $i$  和节点  $j$  之间的边权参数， $r_{ij}$  的值越接近 1，说明两个交叉口关联程度越高。

由于引入了交叉口关联程度  $r_{ij}$ ，改进后的元素  $e_{ij}$  表示为公式 (3-11)：

$$e_{ij} = \begin{cases} \frac{r_{ij}}{\sum_{i,j} r_{ij}}; & \text{节点 } i \text{ 与节点 } j \text{ 之间有边相连} \\ 0; & \text{其他} \end{cases} \quad (3-11)$$

$a_i$  可以表示为公式 (3-12)：

$$a_i = \frac{\sum_j e_{ij}}{\sum_{i,j} r_{ij}} \quad (3-12)$$

改进后的 Newman 快速划分算法在进行社团的合并时，仍然选择能使模块度增量  $\Delta Q$  最大的方向进行合并，当只剩一个社团时停止合并，然后选取模块度  $Q$  最大的划分结果作为输出。

改进后的 Newman 快速算法的执行的步骤如下：

步骤 1：依据路网中的交通信息，计算各交叉口之间的关联程度  $r_{ij}$ 。

步骤 2：进行初始划分，将每个节点视为一个社团，模块度  $Q=0$ ，使用公式 (3-11)

(3-12) 初始化矩阵 E 和数组 A 中的所有元素  $e_{ij}$ 、 $a_i$ 。

$$e_{ij} = \begin{cases} \frac{r_{ij}}{\sum_{i,j} r_{ij}}; & \text{节点 } i \text{ 与节点 } j \text{ 之间有边相连} \\ 0; & \text{其他} \end{cases}$$

$$a_i = \frac{\sum_j e_{ij}}{\sum_{i,j} r_{ij}}$$

步骤 3：选择两个有边相连的社团进行合并，使用公式 (3-9) 计算合并前后模块度的增量  $\Delta Q$ 。

$$\Delta Q = e_{ij} + e_{ji} - 2a_i a_j = 2(e_{ij} - a_i a_j)$$

选择  $\Delta Q$  最大的社团进行合并，更新矩阵 E 和模块度 Q。

步骤 4：重复步骤 3，直至网络中的全部节点合并为 1 个社团，算法结束，选择最大 Q 值对应的划分作为路网动态控制子区划分的最终结果。

### 3.4 基于改进 Newman 快速算法的动态控制子区划分建模

对路网进行动态控制子区划分，以各个子区为控制对象，可以有效降低协同控制的复杂度，提高控制效率。由于交通路网是一种复杂网络，因此可以使用改进后 Newman 快速算法对路网中的交叉口进行子区划分。

首先对路网结构进行抽象，将路网中的交叉口抽象为点，将路网中的道路抽象为边，路网将被抽象为一个复杂网络的拓扑结构图。然后，由于路网中交通流的复杂性、时间相关性、不确定性，路网中的交叉口之间不能简单定义为布尔关系，而是应当依据相邻交叉口之间的路网结构和交通流特性，根据公式 (3-10) 相邻交叉口关联程度。最后使用改进后 Newman 快速算法对路网进行动态控制子区划分。

交通路网动态控制子区划分的具体步骤如下：

步骤 1：对路网结构进行抽象，交叉口抽象为点，道路抽象为边，得到路网的拓扑结构图。

步骤 2：对路网中的交叉口进行编号，收集各相邻交叉口的间距、信号周期时长、交通流量等信息。

步骤 3：使用改进后的 Newman 快速算法对路网进行子区划分。

步骤 4：选择最大 Q 值对应的划分当做路网动态控制子区划分的最终结果。

## 3.5 实验设计及结果分析

### 3.5.1 实验设计

GLD(green light district)是一款开源的微观交通仿真软件,可以自由创建和编辑路网地图,对各种交通信号控制方法进行仿真和评估。为了验证本文方法在交通路网控制子区划分的有效性,将改进后的 Newman 快速算法在 GLD 仿真平台中进行实验和仿真。图 3.2 为仿真的交通路网图,图 3.3 为路网的简化图:

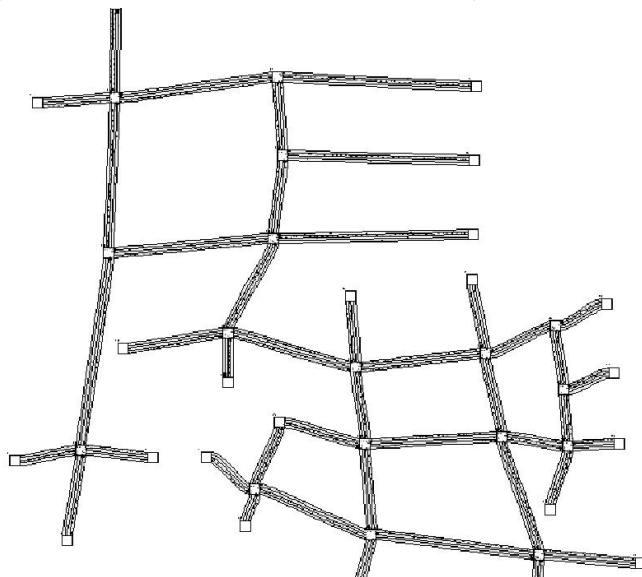


图 3.2 仿真的交通路网

图中包含了十字交叉口、T 字交叉口、直角交叉口等多种交叉口,合计 39 个。该交通路网的参数如下:

**边缘交叉口:** 图中共有 21 个边缘交叉口。边缘交叉口既是车辆的起点,也是车辆的终点。每个边缘交叉口随机生成车辆,每辆车随机选择一个其他边缘交叉口作为终点。

**交叉口:** 图中共有 18 个交叉口,其中 17 个上有交通信号灯,1 个没有交通信号灯。

**车道:** 图中共有 43 条道路,每条道路包含 4 条车道,连接 2 个交叉口,共有 172 条车道。每条车道长短不一,能容纳 10 至 40 辆车左右。

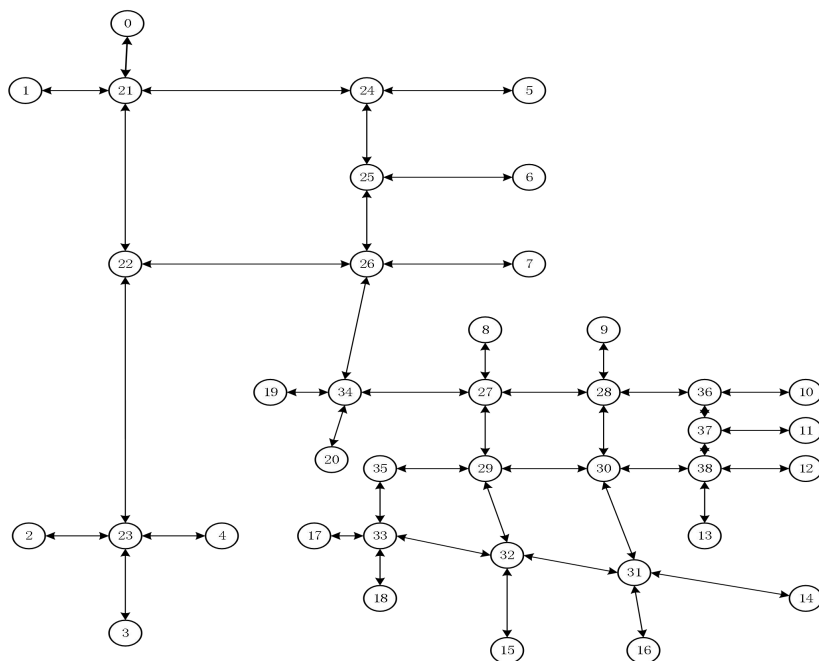


图 3.3 路网的简化图

首先将网络视为无权网络，计算交叉口间的关联矩阵，使用传统 Newman 快速算法对路网进行划分。然后考虑各交叉口之间的关联程度，使用改进后的 Newman 快速算法对网络进行划分。最后将划分结果进行对比分析。

### 3.5.2 结果分析

传统 Newman 快速算法划分的交通控制子区结果如图 3.4 所示，子区划分结果如图 3.5 所示，划分过程模块度中的模块度变化如图 3.6 所示。

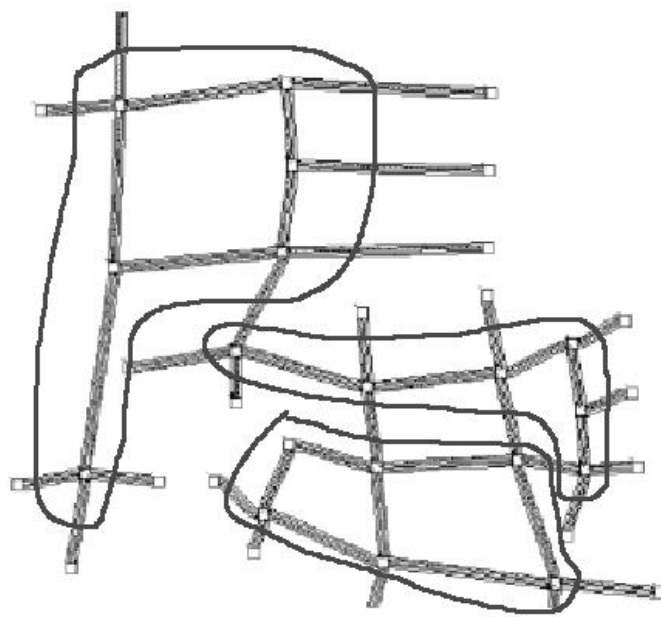


图 3.4 传统 Newman 快速算法交通控制子区



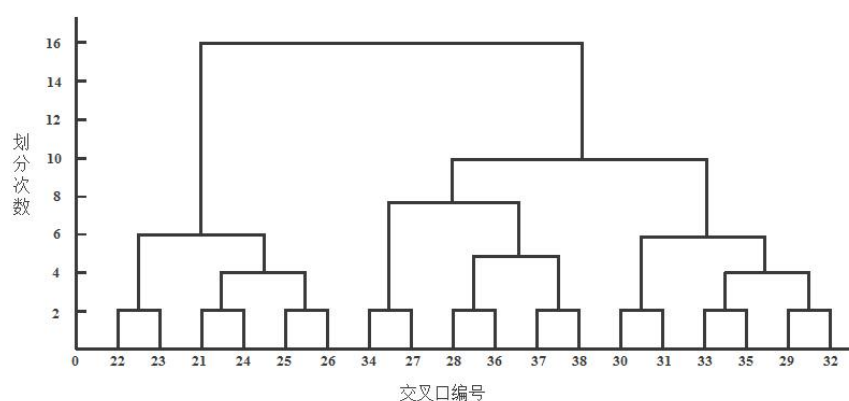


图 3.5 传统 Newman 快速算法子区划分结果

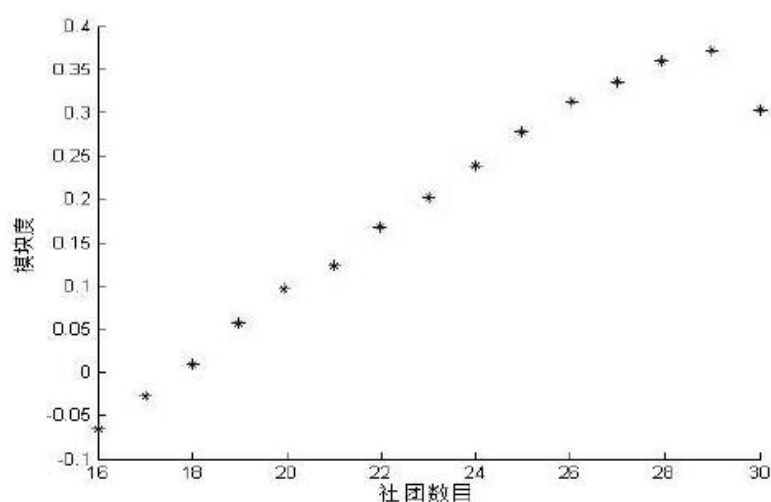


图 3.6 传统 Newman 快速算法子区划分模块度

统计路网中相邻交叉口间距、信号周期时长、交通流量、排队长度和道路密度信息，计算得到的关联程度如表 3.1。使用改进后的 Newman 快速算法划分的交通控制子区结果如图 3.7 所示，子区划分结果如图 3.8 所示，划分过程模块度中的模块度变化如图 3.9 所示。

通过图 3.6 可以发现，传统 Newman 快速算法的模块度  $Q$  的最大值为 0.3652，此时子区划分结果最优，路网被划分为 3 个控制子区。其中 21、22、23、24、25、26 为一个控制子区，29、30、31、32、33、35 为一个控制子区，27、28、34、36、37、38 为一个控制子区。

通过图 3.9 可以发现，改进后的 Newman 快速算法的模块度  $Q$  的最大值为 0.5075，此时子区划分结果最优，路网被划分为 6 个控制子区。其中 21、22、24、25、26 为一个控制子区，23 为一个控制子区，34 为一个控制子区，27、28 为一个控制子区，36、37 为一个控制子区，29、30、31、32、33、35、38 为一个控制子区。

表 3.1 相邻交叉口关联程度

路段名称	关联程度	路段名称	关联程度
21-24	0.883	21-22	0.562
22-26	0.875	22-23	0.231
34-27	0.364	24-25	0.702
27-28	0.841	25-26	0.757
28-36	0.472	26-34	0.346
29-35	0.764	27-29	0.501
29-30	0.596	28-30	0.554
30-38	0.869	36-37	0.906
33-32	0.696	37-38	0.463
32-31	0.892	33-35	0.884
29-32	0.711	30-31	0.557

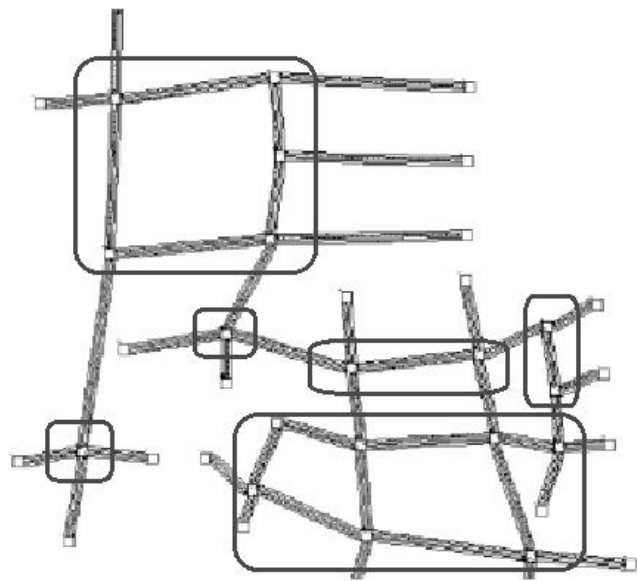


图 3.7 改进后的 Newman 快速算法交通控制子区组成

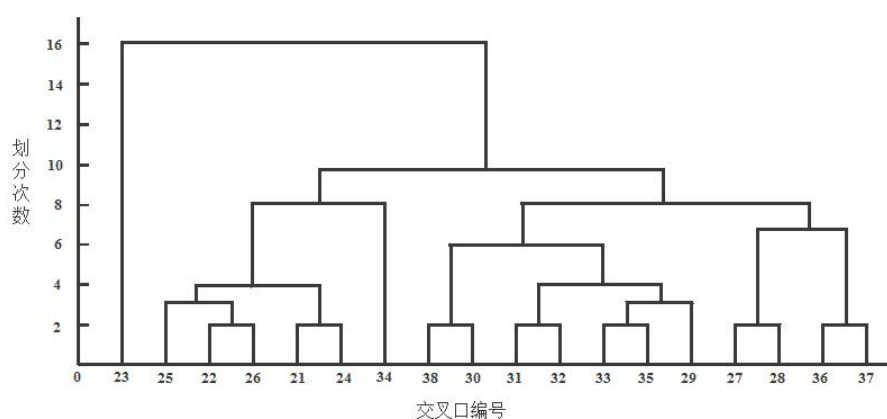


图 3.8 改进后的 Newman 快速算法子区划分结果

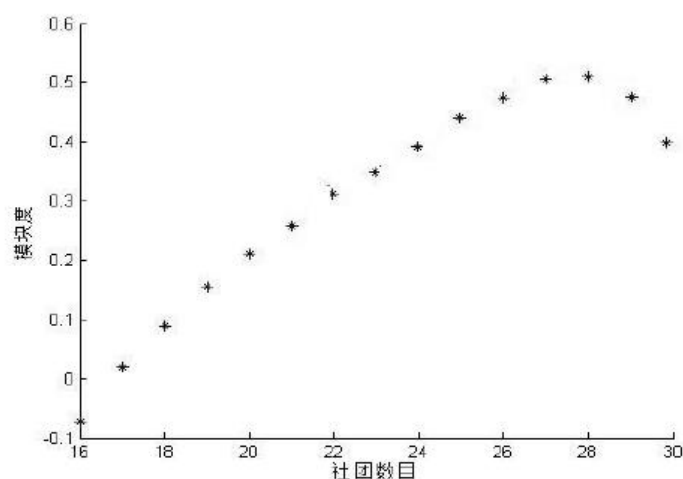


图 3.9 改进后的 Newman 快速算法子区划分模块度

实验中，由于设置第 23 号交叉口与邻近交叉口的距离极长，34 号交叉口的交通状况与其他交叉口不相似，因此，23、34 号交叉口与其他交叉口之间的相似程度较低，不具有和其他交叉口协同控制的必要。改进后的 Newman 快速算法成功将两交叉口单独划分，而传统 Newman 快速算法由于仅考虑道路的拓扑结构，不能进行有效划分。

由于各交叉口的交通流量、排队长度和道路密度、信号周期都随时间发生变化。因此，1000 个仿真周期后，使用改进后的 Newman 快速算法再次对图 3.2 所示的路网进行子区划分。计算得到的关联程度如表 3.2。使用改进后的 Newman 快速算法划分的交通控制子区结果如图 3.10 所示，子区划分结果如图 3.11 所示，划分过程模块度中的模块度变化如图 3.12 所示。

表 3.2 相邻交叉口关联程度（第二次）

路段名称	关联程度	路段名称	关联程度
21-24	0.869	21-22	0.583
22-26	0.851	22-23	0.254
34-27	0.684	24-25	0.715
27-28	0.863	25-26	0.762
28-36	0.464	26-34	0.335
29-35	0.792	27-29	0.506
29-30	0.603	28-30	0.503
30-38	0.347	36-37	0.889
33-32	0.702	37-38	0.723
32-31	0.867	33-35	0.872
29-32	0.754	30-31	0.573

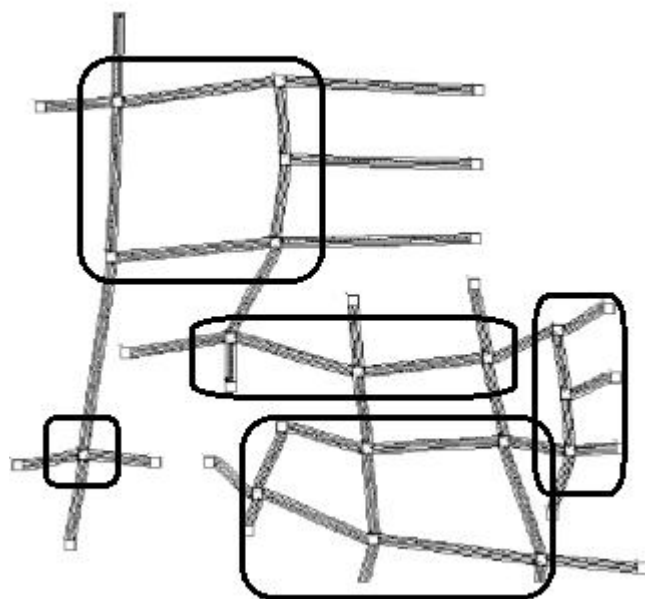


图 3.10 改进后的 Newman 快速算法交通控制子区组成（第二次）

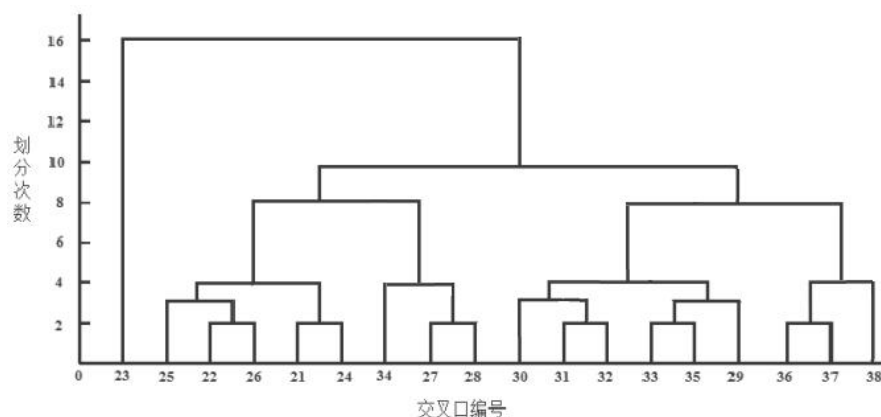


图 3.11 改进后的 Newman 快速算法子区划分结果（第二次）

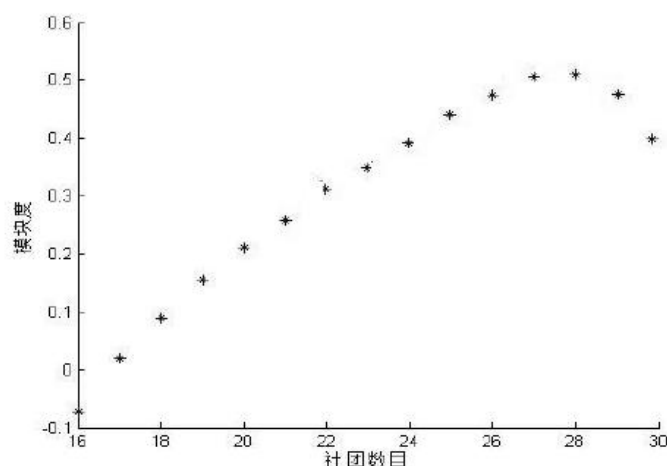


图 3.12 改进后的 Newman 快速算法子区划分模块度（第二次）

由图 3.11、图 3.12 可知，各交叉口的交通流量、排队长度和道路密度、信号周期的变化、会对子区划分结果产生一定的影响。路网被划分为 5 个控制子区。其中 21、22、24、25、26 为一个控制子区，23 为一个控制子区，34、27、28 为一个控制子区，36、37、38 为一个控制子区，29、30、31、32、33、35 为一个控制子区。由图 3.12 可知，改进后的 Newman 快速算法能够适应交通流的变化，将关联程度强的交叉口划分到一个子区，将关联程度弱的交叉口划分到不同子区。

对比两种算法的子区划分结果，可以发现改进后的 Newman 快速算法的模块度更高，对路网进行子区划分的结果更加准确和细致，符合交通流的特性。同时，由于路网的拓扑结构一般不会发生变化，所以传统 Newman 快速算法对路网的划分结果是固定的，不能实现对路网的动态划分。而改进后的 Newman 快速算法考虑交叉口之间的关联特性，相邻交叉口间距、信号周期时长、交通流量、排队长度和道路密度等因素，能够实时的根据路网状态进行划分，使划分结果更加符合交通流的特性，且能够实现

路网控制子区的动态划分。

### 3.6 本章小结

本章提出了一种改进的 Newman 快速划分算法，并应用于交通路网的划分。首先介绍了社团结构的定义和几种常见的社团划分算法。然后给出了交叉口关联程度的计算方法，之后提出改进的 Newman 快速划分算法并应用于交通路网的划分。最后使用 GLD 软件进行对比实验，验证改进后算法的性能。实验结果表明，改进后的算法能够对区域路网进行合理、正确的动态划分，且划分结果更加符合交通流的特性。

## 第四章 基于 CNN 和 GRU 网络的交通流预测方法

交通流预测在城市智能交通系统中占有重要地位,在现有的多种交通流预测模型中,仅考虑交通流的时间特征,缺乏对交通流空间特征的考虑。因此,本章提出了一种基于 CNN 和 GRU 网络的交通流预测模型。分别使用 CNN 和 GRU 神经网络对交通流的空间特征和时间特征进行提取。本章的主要内容包括对交通流预测问题描述,模型的构建过程,实验设计和结果分析。

### 4.1 交通流预测问题描述

交通流量是评价道路通行能力的重要标准,如果能够预测未来一段时间内的交通流量,就能够更好的对路网进行控制,降低发生交通拥堵的可能。交通流预测问题可以描述如下:

在包含  $N$  个路段的路网中,路段  $i$  的交通流量序列表示为  $X^i = (x_t^i, x_{t+1}^i, \dots, x_{t+h}^i)$ ,  $1 \leq i \leq N$ , 其中,  $x_t^i$  表示在第  $t$  个时间间隔内路段  $i$  的交通流量,通常以 5 到 15 分钟作为一个时间间隔。交通流预测就是通过使用前  $r$  个时间间隔内的交通流数据来预测第  $h$  个时间间隔后的交通流量  $x_{t+h}$ 。  $N$  个路段的时空流量集合矩阵表示为:

$$X_t = \begin{bmatrix} x_{t-r+1}^1 & x_{t-r+1}^2 & \cdots & x_{t-r+1}^N \\ x_{t-r+2}^1 & x_{t-r+2}^2 & \cdots & x_{t-r+2}^N \\ \vdots & \vdots & \ddots & \vdots \\ x_t^1 & x_t^2 & \cdots & x_t^N \end{bmatrix} \quad (4-1)$$

预测的交通流数据表示为:

$$X_{t+h} = \begin{bmatrix} x_{t+h}^1 \\ x_{t+h}^2 \\ \vdots \\ x_{t+h}^N \end{bmatrix} \quad (4-2)$$

#### 4.1.1 交通流预测模型

交通流预测模型的流程图如图 4.1 所示:

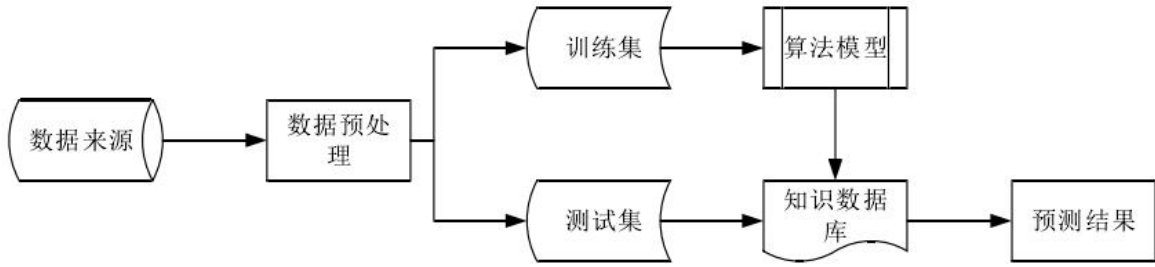


图 4.1 交通流预测模型流程图

交通流预测模型的具体步骤如下：

(1) 数据来源：可以使用车辆探测器采集交通流量、车辆速度、道路占有率等多种交通流数据。目前我国常见的探测器有视频探测器、超声波探测器、磁感探测器和微波探测器等。此外，也可以通过车辆的 GPS 信息、电子车牌等收集数据。

(2) 数据预处理：数据预处理的情况对交通流预测模型的准确度具有重要影响。首先，根据数据采集时间间隔，对数据集中存在明显误差的数据进行清理。然后，对于缺失数据，使用拉格朗日插值法进行补全。最后，由于不同路段，交通流差异较大，因此需要对数据进行归一化处理。在各种常用的归一化方法中，本文选择最小最大标准化方法，计算方法见公式（4-3）

$$x' = \frac{x - \min X}{\max X - \min X} \quad (4-3)$$

其中， $x$  表示交通流量， $\min X$  表示交通流量中的最小值， $\max X$  表示交通流量中的最大值。

将处理完成后的流量信息以矩阵的形式进行保存。

(3) 数据集构建：将处理后的数据组成数据集，并分为训练集和测试集。

(4) 算法模型：针对交通流的不确定性、时间相关性、非线性等特征，选择一种合适的模型，使用训练集对模型进行训练，更新并保存模型参数，建立交通流预测知识数据库。

(5) 结果评估：使用测试集数据对交通流预测模型进行测试，并对预测结果进行评估。

#### 4.1.2 交通流预测的误差评估

在对交通流进行预测时，需要尽最大可能的保证预测模型的准确性。但是，由于交通流固有的不确定性、非线性等特征，导致预测结果与实际结果必然存在误差，因此，选择合适的评价指标对误差进行分析，是评估预测模型性能的重要组成部分。

在现有的研究中，主要采用以下三种评价指标：

1. 均方根误差（Root Mean Squared Error, RMSE）



$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4-4)$$

2. 平均绝对误差 (Mean Absolute Error, MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4-5)$$

3. 相对误差 (Relative Error, RE)

$$RE = \frac{|y_i - \hat{y}_i|}{|y_i|} \times 100\% \quad (4-6)$$

其中,  $y_i$  表示真实流量值, 而  $\hat{y}_i$  表示算法输出的预测流量值,  $n$  为样本数量。本文采用均方根误差对模型的性能进行评估。

## 4.2 基于 CNN 和 GRU 网络的交通流预测方法建模

本文所设计的基于 CNN 和 GRU 网络的交通流预测模型, 通过对交通流的空间特征和时间特征进行提取和预测, 将预测结果进行整合后, 得到预测模型的输出结果。空间特征的提取是指提取同一时刻不同位置上的信息, 时间特征的提取是指提取同一位置在不同时刻的信息。模型的基本框架如图所示:

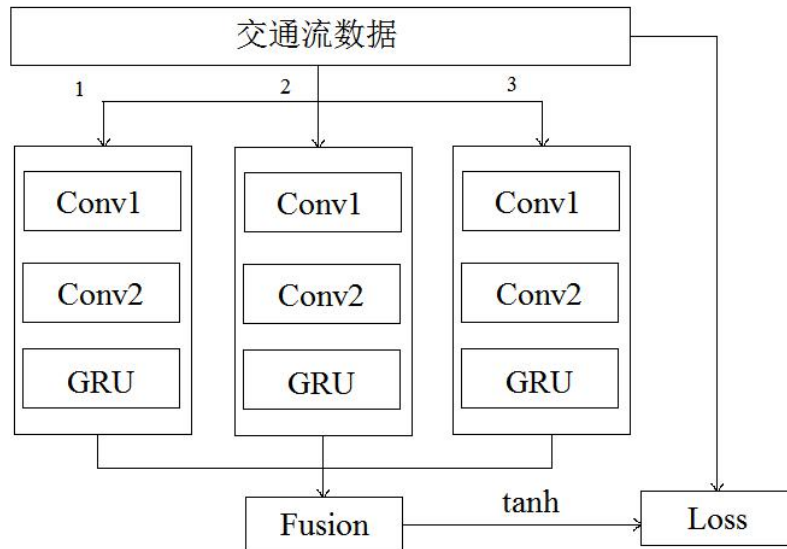


图 4.2 模型的基本框架

### 4.2.1 使用 CNN 网络对交通流空间特征进行提取及预测

提取空间特征就是通过同时提取不同位置上的信息数据, 发现空间位置信息与数据之间存在的关系。由于交通流具有空间依赖性, 当前道路的流出流量, 将对下游的相邻道路甚至更远处道路的交通流产生影响, 同时当前道路的流入流量, 也受到上游

相邻道路甚至更远处道路的交通流影响。因此，考虑路网交通流数据的空间特征，对于交通流的预测有重要意义。

CNN 网络能够发现数据的局部和整体之间的关系，能够捕捉空间结构信息。因此，本节使用 CNN 网络对交通流量的空间信息进行提取。CNN 网络结构如图 4.3 所示。

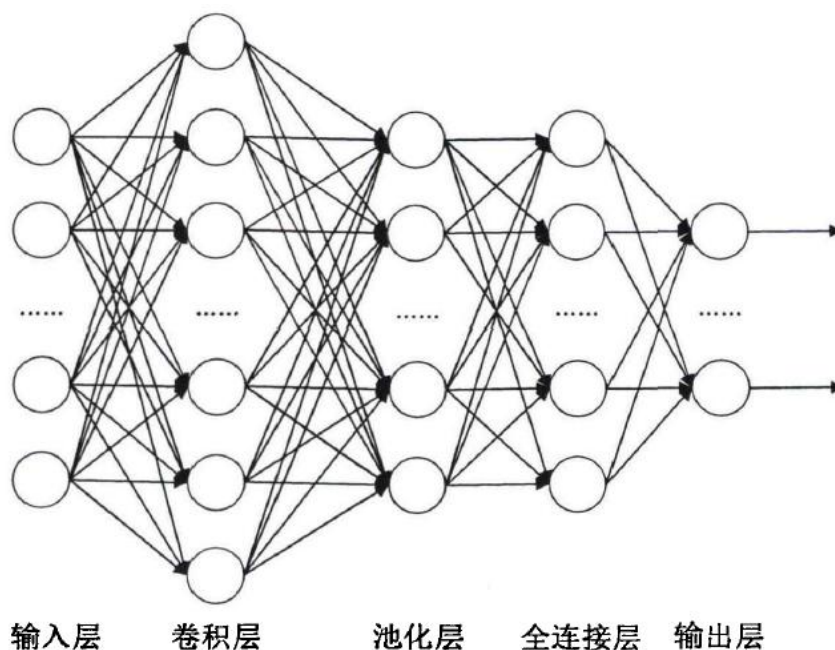


图 4.3 CNN 网络结构图

在卷积层中需要设置卷积核的大小，卷积核的大小将对模型的性能产生影响。当卷积核的尺寸较小时，容易提取局部特征，但可能忽视局部特征之间的关联性；当卷积核的尺寸较大时，容易忽视局部特征，同时增加计算量。因此，卷积核大小通常设置为  $3 \times 3$  或  $5 \times 5$ 。在 CNN 网络中使用 dropout 方法缓解过拟合，损失函数使用平方误差损失函数，使用批量梯度下降算法进行模型的优化。

#### 4.2.2 使用 GRU 网络对交通流时间特征进行提取及预测

时间特征的提取是提取同一位置在不同时间上的信息。发现时间信息与数据之间存在的关系。由于交通流具有时间依赖性，表现为同一天内，某一路段在  $t$  时刻的交通流，将受到当前路段在  $t-1, t-2, \dots$  时刻的交通流影响，且间隔越长影响越小。同时工作日之间交通流变化相似，具有一定的周期性，如早晚高峰的峰值、时间点基本相同。图 4.4 为某一路段连续 3 天的交通流量。因此，考虑路网交通流数据的时间特征，对于交通流的预测有重要意义。

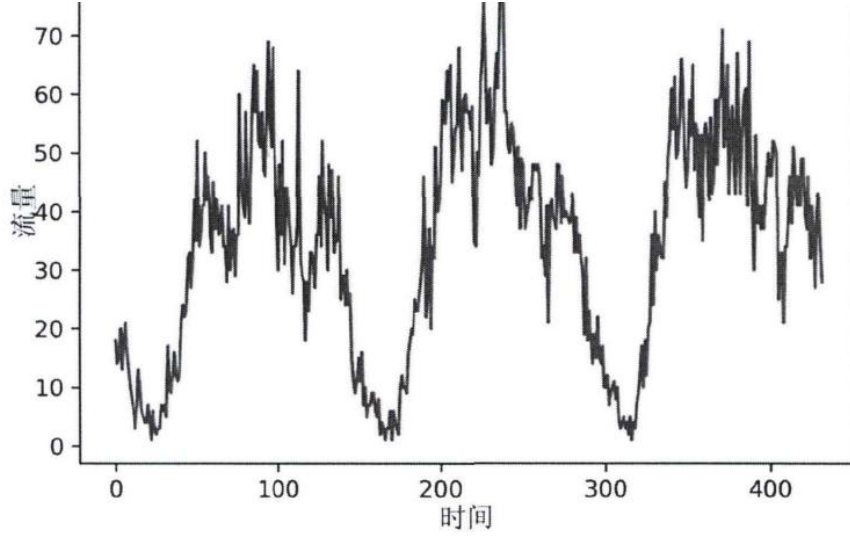


图 4.4 某一路段连续 3 天的交通流量

GRU 网络能够发现数据间的时间关系，利用之前的时间序列数据预测下一时刻的数据，解决了传统 RNN 循环神经网络的梯度消失、梯度爆炸等问题，较 LSTM 神经网络减少了参数的数量，增加了计算速度。因此，本节在 CNN 网络之后组合 GRU 网络，使用 GRU 网络对交通流量的时间信息进行提取。

整个预测模型包含三个相似的网络结构，每个网络结构的输入数据拥有不同的时间跨度和步长，能够从三个角度分析交通流的时间依赖性。第一部分考虑交通流的临近影响，输入数据的时间跨度为 10 分钟，即 1 个时间间隔，步长为 8；第二部分考虑交通流的周期影响，输入数据的时间跨度为 1 天，即 144 个时间间隔，步长为 4；第三部分考虑交通流的趋势影响，输入数据的时间跨度为 5 天，即 720 个时间间隔，步长为 2。

### 4.2.3 预测结果整合

将三个网络结构的预测结果进行融合，得到最终的预测结果。计算方法为：

$$\hat{y} = \sigma(w_1 \hat{y}_1 + w_2 \hat{y}_2 + w_3 \hat{y}_3) \quad (4-7)$$

其中， $w_1$ ， $w_2$ ， $w_3$  为各部分的权重， $\hat{y}_1$ ， $\hat{y}_2$ ， $\hat{y}_3$  为各部分的预测结果， $\sigma$  为  $\tanh$  激活函数。

### 4.2.4 参数选择

神经网络模型中参数的设置对预测结果的准确性有重要影响。因此，在基于 CNN 和 GRU 网络的交通流预测模型中需要对网络的各种参数进行设置。模型中的参数可以分为模型自学习参数和超参数。

模型自学习参数在交通流预测模型的网络结构、激活函数、学习率等参数设定完

毕后，会随着训练过程自动调整，不需要人为干预，直到网络达到最大训练次数或满足精度要求。主要包括网络内部的权重矩阵、偏置项等

超参数需要提前进行设置，主要包括网络的层数、每层节点数目、激活函数、损失函数、学习率、训练次数以及卷积核的个数、大小、步长等。超参数的设置一般根据经验和测试来调整，没有统一的方法。

基于 CNN 和 GRU 网络的交通流预测模型中包含三个相似的网络结构，每个网络结构中包括两个 CNN 网络和一个 GRU 网络。在每个网络结构中，将第一个 CNN 中卷积核个数设置为 32，大小为 3\*3，步长为 1，第二个 CNN 中卷积核个数设置为两倍的输入数据的步长，大小为 3\*3，步长为 1，使用 Sigmoid 和 Tanh 作为激活函数，最大训练次数 5000，学习率为 0.02，批度大小为 32，优化器为 Adam，dropout 大小设置为 0.2。使用单层 GRU 神经网络，神经元个数为 60。

## 4.3 实验设计及结果分析

### 4.3.1 实验设计

本章中所使用的路网交叉口历史流量数据来自于 2015 年 4 月 1 日至 2015 年 6 月 30 日工作日期间北京市出租车的真实历史交通数据，数据为 10 分钟内通过各路口的车辆总数，如表 4.1 所示。神经网络参数设置见 4.2.4。

表 4.1 交通流量时间序列

时间	交通流量（辆）
...	...
2015-5-2 9:00:00	39
2015-5-2 9:10:00	52
2015-5-2 9:20:00	81
2015-5-2 9:30:00	63
2015-5-2 9:40:00	85
...	...

### 4.3.2 结果分析

基于 CNN 和 GRU 网络的交通流预测方法实验结果如图 4.5 所示, LSTM 算法的实验结果如图 4.6 所示, 不同模型的预测误差如表 4.2 所示。

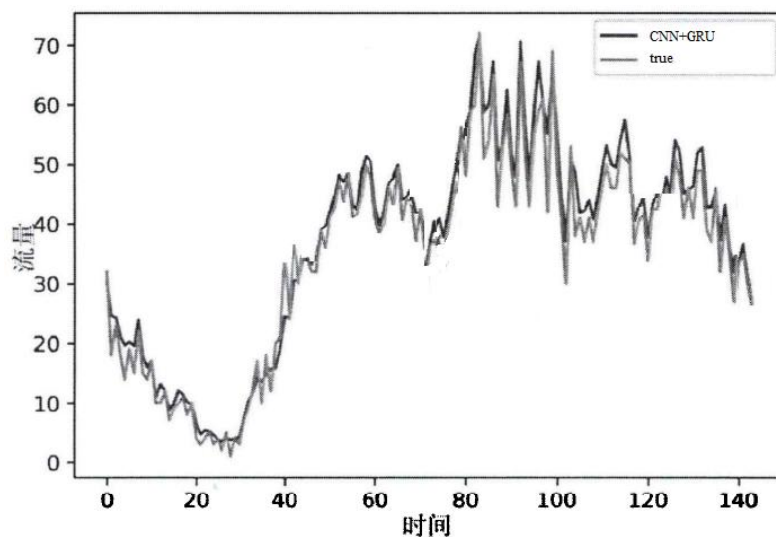


图 4.5 CNN+GRU 预测值与真实值对比

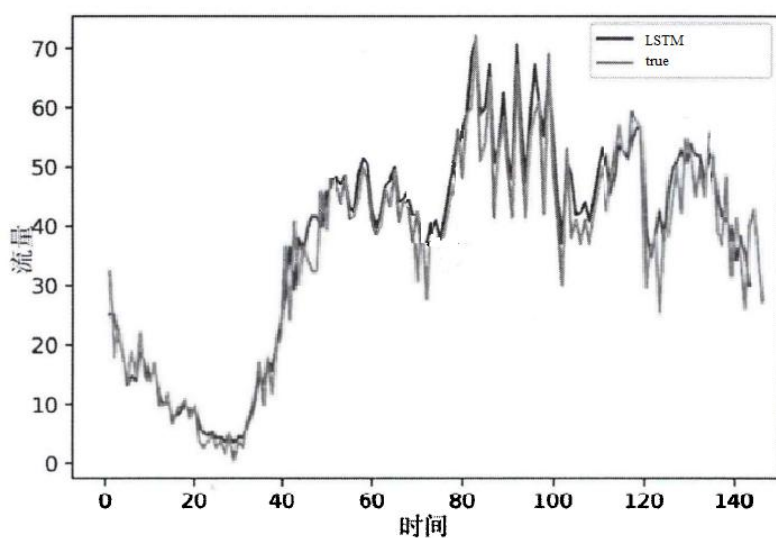


图 4.6 LSTM 预测值与真实值对比

表 4.2 不同模型的预测误差

模型名称	评价指标 (RMSE)
CNN+GRU	4.25
LSTM	6.72

由表 4.2 可知，基于 CNN 和 GRU 网络的预测算法与基于 LSTM 的预测算法都能完成对交通流量的预测，但在预测结果上，基于 CNN 和 GRU 网络的预测算法与基于 LSTM 的算法相比，具有较小的均方根误差（RMSE），能够对交通流量进行更加准确的预测。因此，基于 CNN 和 GRU 网络的交通流预测算法通过综合考虑交通流的空间特征和时间特征，能够很好地完成对交通流量的预测。

## 4.4 本章小结

本章首先描述了交通流预测问题，介绍了交通流预测模型和评估误差。然后针对交通流的空间特征和时间特征，分别使用 CNN 和 GRU 神经网络进行特征提取。提出了结合 CNN 和 GRU 网络的交通流预测模型，并使用真实的交通流数据进行测试。实验表明，结合 CNN 和 GRU 网络的交通流预测模型能够对交通流量进行快速、准确的预测。

## 第五章 结合流量预测的 Double DQN 交通灯区域控制方法

对交叉口的交通灯控制方法进行合理优化能够提升城市交通体系的运行效率。在现有的交通灯控制算法中，没有很好的解决多交叉口协同控制的问题，且控制策略具有一定的滞后性，因此本章提出了一种基于深度强化学习交通灯区域控制方法。使用子区划分算法决定交通灯协同控制的区域范围，使用流量预测算法提高控制策略的前瞻性，基于 Double DQN 算法实现对多交叉口的协同控制，降低了车辆的等待时间，保障路网的畅通。本章的主要内容包括对交通灯控制问题描述、模型的构建过程、实验设计和结果分析。

### 5.1 交通灯控制问题描述

交通灯控制对于缓解城市道路交通问题具有重要意义，合理的交通灯控制策略能够有效解决城市道路拥堵问题。通过改变交叉口处交通信号灯的配时控制参数，可以有效降低路网内车辆的等待时间，提高路网的通行能力。

#### 5.1.1 交叉口通用数据描述

本文采用基于路网中车辆状态的描述方法进行建模，在这种建模方法中，状态的数量与车道呈线性关系，因此容易扩展到大型网络中。这种建模方法能够描述车辆的各种信息，包括车辆的位置，车辆的速度、车辆的终点等。与基于交通灯状态的描述方法相比，具有更小的状态空间，函数收敛速度较快。由于现实生活中，黄灯仅作为红灯和绿灯动作变化时的缓冲，且一般设置为固定的时间，不影响红灯和绿灯的动作变化，不能带来控制策略上的收益，所以在本文的控制方法建模过程中，交通灯的状态只包含红灯和绿灯，不考虑黄灯的影响。

路网中的每个交叉口都包含一个信号控制单元  $Agent$ ，整个路网组成了一个多  $Agent$  系统。具体描述如下：

(1)  $i$ ：代表路网中交叉口  $i$  的信号控制单元  $Agent_i$ 。在包含  $n$  个交叉口的路网中，交叉口控制单元的集合为  $I = \{Agent_0, \dots, Agent_i, \dots, Agent_n\}$ 。图 5.1 所示为路网中的一个交叉口。

(2)  $A_i$ ：代表路网中交叉口信号控制单元  $Agent_i$  的相位集合。

(3)  $a_i$ ：代表路网中交叉口信号控制单元  $Agent_i$  在某个时刻的相位动作， $a_i \in A_i$ 。定义  $a_i$  为交叉口中可以同时执行绿灯的交通灯组合，执行动作  $a_i$  就是将交叉口中对应相位方案中的交通灯设置为绿灯，而其他的交通灯设置为红灯。表 5.1 列出

了图 5.1 中交叉口  $i$  的相位与动作对应关系。

(4)  $L$  : 代表路网中全部车道的集合。 $L_i$  代表交叉口  $i$  的所有驶入车道的集合,  $L_i \in L$ 。 $l_i$  代表交叉口  $i$  的其中一个驶入车道,  $l_i \in L_i$ 。

(5)  $TL$  : 代表路网中所有信号灯的集合。 $TL_i$  代表路网中交叉口  $i$  的  $L_i$  对应的信号灯组成的集合,  $tl_i$  代表驶入车道  $l_i$  上的信号灯。

(6)  $tla$  : 代表路网中所有信号灯  $tl$  的动作集合。信号的动作于信号灯的状态一一对应, 信号灯包含红、绿两种颜色, 动作集合为  $tla = \{tla_{green}, tla_{red}\}$ 。

(7)  $s_{pl_i}$  : 由三元组  $[tl_i, pos, des]$  组成,  $pos$  表示车辆在当前时间步所处的位置,  $des$  表示车辆行驶的终点,  $tl_i$  表示车辆所在车道上的交通灯。

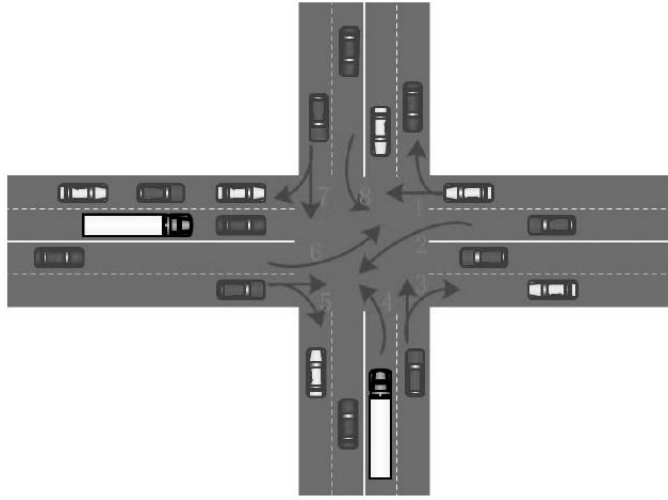


图 5.1 路网中的一个交叉口

表 5.1 交叉口相位与动作对应关系

交叉口动作 ( $a_i$ )	相位 (id)	
$a_0$	1	2
$a_1$	3	4
$a_2$	5	6
$a_3$	7	8
$a_4$	1	5
$a_5$	3	7
$a_6$	2	6
$a_7$	4	8



以上为通用的数据描述，具体的交通信号灯控制方法的建模见 5.3。

### 5.1.2 交叉口控制算法性能评价指标

为了判断路网的拥堵情况，对各个交通信号灯控制算法进行对比和评价，采用交叉口平均等待时间、车辆平均行驶等待时间、已经到达终点的车辆总数三个指标对交叉口性能进行评价：

(1) 交叉口平均等待时间 AJWT (Average Junction Waiting Time)：使用路网中全部车辆通过交叉口时的等待时间之和除以车辆总数，反映了交叉口前单个车辆的平均等待时间。AJWT 值越小，说明车辆在交叉口的等待时间越少，算法性能越好；AJWT 值越大，说明车辆在交叉口的等待时间越长，算法性能越差。

(2) 车辆平均行驶等待时间 ATWT (Average Trip Waiting Time)：使用全部抵达终点的车辆的合计等待时间除以抵达终点的车辆的数量，反映了一辆车在整个行程中的平均等待时间。ATWT 值越小，说明车辆等待时间越少，算法性能越好；ATWT 值越大，说明车辆的等待时间越长，算法性能越差。

(3) 已经到达终点的车辆总数 TVA (Total Vehicles Arrived)：表示路网中已经到达目的地的车辆总数。TVA 值越大，说明到达目的地车辆数越多，算法性能越好；TVA 值越小，说明到达目的地车辆数越少，算法性能越差。

## 5.2 结合流量预测的 Double DQN 算法

深度强化学习算法 (Deep Reinforcement Learning, DRL) 是一种将强化学习算法 (Reinforcement Learning, RL) 与深度学习算法 (Deep Learning, DL) 相结合的算法，能够有效解决状态空间连续或者较大时引发的“维度灾难”，被广泛应用在各种领域，DQN (Deep Q-Network)<sup>[58]</sup> 算法在其中占有重要地位。由于 DQN 的出色表现，研究者基于 DQN 进行了多种改进和扩展，常见的有双 DQN (Double DQN)<sup>[59]</sup>、对抗 DQN (Dueling DQN)<sup>[60]</sup>、分布式 DQN (Distributional DQN) 等，对于这些改进算法的研究提高了神经网络的稳定性，在各个领域得到了广泛应用。

DQN 是在 Q-Learning 的基础上，结合神经网络改进得到的，通过引入目标网络得到标签用以监督训练评估网络，通过采用经验回放机制 (Reply Experience) 打破样本之间的相关性。在每次训练后，根据损失函数使用梯度下降算法对目标网络和评估网络的参数进行更新。DQN 算法计算 Q 值方法如式 (5-1)：

$$Y_t^{DQN} = R_{t+1} + \gamma Q(s_{t+1}, \arg \max_a Q(s_{t+1}, a; \theta^-); \theta^-) \quad (5-1)$$

其中  $R$  代表即时奖励， $\gamma$  代表折扣因子， $\theta$  代表网络的参数。DQN 的主要框架如图 5.2 所示，通过建立回放经验池，当智能体与环境发生交互时，将当前的状态、

动作、奖励值和下一步的状态作为一个数据样本存储在回放经验池，训练时每次随机均匀抽取部分数据。这种机制能够有效打破样本之间的相关性，平滑数据分布，同时提高样本的利用率。在训练过程中，将数据样本作为目标网络的输入，选择执行能够产生最大 Q 值的动作，生成评估网络中监督学习的标签。训练过程中利用损失函数对网络的性能进行评估，并采用反向传播算法对目标网络和评估网络的参数进行更新。

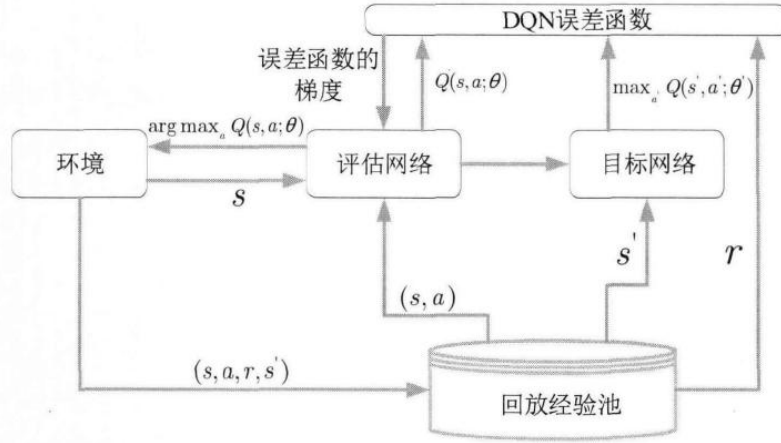


图 5.2 DQN 算法框架

在 DQN 中，目标网络的动作通常选择能够产生最大 Q 值的动作，这种通过贪心策略得到 Q 值会引发整个网络过度估计的问题。Double DQN 算法使用了两个独立的网络，将动作的选择和评估进行拆分，解决了过度估计的问题，提高了网络的性能。Double DQN 算法计算 Q 值方法如式（5-2）：

$$Y_t^{Double\ DQN} = R_{t+1} + \gamma Q(s_{t+1}, \arg \max_a Q(s_{t+1}, a; \theta_t^-); \theta_t^-) \quad (5-2)$$

Double DQN 改变了公式（5-1）中  $\arg \max$  函数中的第三个参数。在每一个训练步中，当目标网络输入下一状态时，不再直接选择使目标网络的输出中最大值对应的动作。而是选择评估网络关于下一状态的输出中最大值对应的动作，然后在目标网络中找到对应动作的输出值。

在使用 Double DQN 算法进行交通信号控制时，控制结果存在一定的滞后性，且不能很好的应对各种突发状况，因此，本文在 Double DQN 算法的基础上，加入预测模块，通过对未来状态的提前判断，能够得到更加符合当前和未来状态的输出结果，提高了应对突发状况的能力。

在结合流量预测的 Double DQN 算法中，在每个训练步，Agent 与环境进行交互，得到当前状态。与此同时，Agent 将当前状态发送到交通流预测模型之中。在交通流预测模型中，依据当前的交通流状态  $s_t$ ，使用公式（5-3）对下一时刻的交通流状态

进行预测，得到下一时刻的预测状态  $s_{t+1}'$ ，并将预测结果  $s_{t+1}'$  返回给 Agent。Agent 计算当前状态和预测将来状态中的最优 Q 值，按照公式 (5-4) 进行动作的选择，并根据状态和动作获得对应的奖励值，通过公式 (5-5) 对 Q 值进行更新。

$$s'_{t+1} = f_t(s_t) \quad (5-3)$$

$$a' = \arg \max_{a'} Q_t(s'_{t+1}, a) \quad (5-4)$$

$$Q_{t+1}(s_t, a_t) = (1 - \alpha) Q_t(s_t, a_t; \theta) + \alpha * (R_{t+1} + \gamma \max_{a'} Q_t(s'_{t+1}, a'; \theta^-)) \quad (5-5)$$

$s_{t+1}'$  是交通流量预测模型在时间步  $t + 1$  预测的交通流量， $a'$  是  $Q_t(s'_{t+1}, a)$  中使 Q 值最大的动作， $\theta$  代表评估网络的参数， $\theta^-$  代表目标网络的参数， $f_t$  为第四章中的预测模型。

在初始化时，智能体随机进行动作的选取，这就是探索（exploration）。随着训练步长的增加，网络逐渐收敛，每个动作对应的 Q 值将更加准确，此时可以优先选取对应 Q 值更大的动作，这就是利用（exploitation）。为了解决过多的利用策略造成的局部最优解的问题，可以使用  $\varepsilon$ -greedy 策略，在进行迭代训练时，以  $\varepsilon$  概率进行探索，以  $1 - \varepsilon$  的概率进行利用。 $\varepsilon$  随迭代次数的增加而减小，计算方法如公式 (5-6)。

$$\varepsilon = \max\left(0.01, 1 - \frac{n}{N}\right) \quad (5-6)$$

其中  $n$  是当前迭代次数， $N$  是总迭代次数。

结合流量预测的 Double DQN 算法的主要框架如图 5.3 所示。

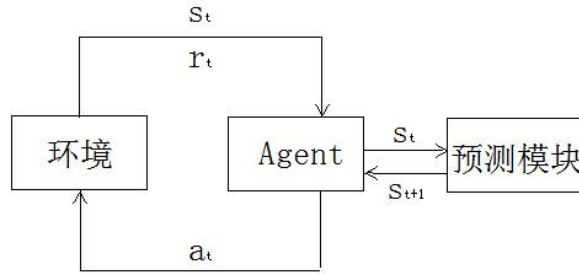


图 5.3 结合流量预测的 Double DQN 算法框架

结合流量预测的 Double DQN 算法如算法 1 所示：

---

算法 1： 结合流量预测的 Double DQN 算法框架

---

输入：初始状态  $s$

输出：执行动作  $a$ ，即 Q 值

初始化评估网络参数  $\theta$  和目标网络参数  $\theta^-$ ，初始化回放经验池  $D$ ，批度大小为  $m$ ，学习率  $\alpha$ ，折扣因子  $\gamma$ ，探索率  $\varepsilon$ ，目标网络更新间隔  $C$ ，训练轮次  $T$ ，当前训

练步  $t=1$ 。

while  $t < T$

观察当前状态  $s_t$ ，使用公式 (5-3) 对当前状态  $s_t$  进行预测，获得下一预测状态  $s_{t+1}'$

将  $s_t$  输入评估网络，得到评估网络中所有动作对应的 Q 值输出，使用  $\varepsilon$ -greedy 策略，在所有的 Q 值输出中选择对应的动作  $a$

执行动作  $a$ ，获得奖励值  $R$ ，获得下一状态  $s_{t+1}$

添加经验  $experience = (s_t, a, R, s_{t+1}, s_{t+1}')$  到经验池 D 中

更新状态  $s_t \leftarrow s_{t+1}$

从回放经验池 D 中抽取  $m$  个数据  $\{s_t^j, a^j, R^j, s_{t+1}^j, s_{t+1}^{j'}\}, j=1,2,\dots,m$ ，计算 Q 值  $y_j$

$$y^j = R^j + \gamma \max_{a'} Q(s_{t+1}^{j'}, \arg \max_{a'} Q(s_{t+1}^{j'}, a; \theta); \theta^-) \quad (5-7)$$

计算损失函数

$$\nabla \theta = \frac{1}{m} \sum_{j=1}^m (y^j - Q(s^j, a^j; \theta))^2 \quad (5-8)$$

通过梯度反向传播更新评估网络参数  $\theta$

if  $t \% C = 1$  then

更新目标网络参数  $\theta^- \leftarrow \theta$

end if

end while

### 5.3 结合流量预测的 Double DQN 交通灯区域控制方法建模

本文将在子区划分算法、结合流量预测的 Double DQN 算法的基础上，提出结合流量预测的 Double DQN 交通灯区域控制方法 TC-TFP (Traffic Controller with Traffic Flow Prediction)，并建立交通信号控制模型，对交通灯进行区域控制。

#### 5.3.1 路网交通状态空间

使用车辆的状态对路网中交通状态进行表示，车辆状态的实时变化可以代表路网状态的动态变化特性。将任意交叉口直接相连的车道从与停车线有一些距离  $length$  开始，按照固定的长度  $b$  分成若干个格子，使用布尔值对格子内车辆的有无情况进行描述，1 代表格子内有车辆、0 代表格子内没有车辆。以布尔值组成一个车辆位置矩阵，若某辆车跨过两个方格，则选择占用比例较大的。因此，可以用 0 和 1 表示路网

中任意交叉口的车辆情况。图 5.4 为一个道路状态的矩阵描述。

本文中使用道路上车辆的数量作为路网交通状态空间。由于路网中的车辆是实时运动的，所以交叉口的交通状态也实时发生变化。

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

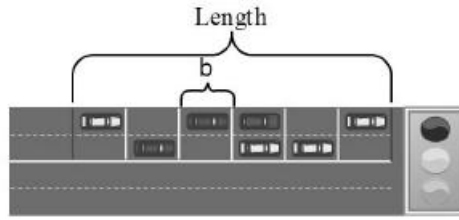


图 5.4 道路状态

### 5.3.2 路网交叉口动作集

信号控制单元 Agent 将依据交叉口的交通状态，在动作空间中进行动作的选择和执行，每次选择最合适的动作来优化交叉口的相位控制。

表 5.1 中列出了所有可能的交通信号控制的动作空间  $\{a_0, a_1, a_2, \dots, a_7\}$ ，由于  $\{a_0, a_1, a_2, a_3\}$  表示的动作每次仅允许来自一个方向的车辆通行，其他三个方向均禁止通行，不符合我国的交通规则，因此实际的动作空间仅包含  $\{a_4, a_5, a_6, a_7\}$ 。交叉口的动作与交通灯相位的实际对应关系如表 5.2 所示。

表 5.2 实际动作空间

交叉口动作 ( $a_i$ )	相位 (id)	
$a_4$	1	5
$a_5$	3	7
$a_6$	2	6
$a_7$	4	8

为了避免控制信号的连续切换，需要定义相位执行的最大时长和最小时长，依据现实中交通信号灯的 average 时长，将任意相位的最大时长设置 70s，最小时长设置为 40s。当交叉口执行某一相位动作时，当下一相位不同时，判断当前相位的执行时间，不足

40s 的应当先执行至 40s 后，再更换相位，超过 40s 的可以立即更换相位；当下一相位相同时，判断当前相位的执行时间，不足 70s 的在该相位执行时长上累加 5s，已达到 70s 的可以立即更换相位。

### 5.3.3 路网交叉口奖励值

当执行动作后，Agent 能够依据环境变化获得对应的奖励，既是对当前策略进行评估，也会影响未来的控制策略。在现有的交通信号控制方法中，有多种类型的指标可以作为奖励值，如交通流量变化、车辆排队长度的变化、车辆等待时间的变化等。应当根据交通控制的目标选择合适的奖励值。

$R$  表示车辆在交通信号灯采取动作后从状态  $s_t$  转移到状态  $s_{t+1}$  的即时奖赏值，如公式 (5-9) 所示。

$$R = \min(q_1, q_2, q_3, q_i) - \max(q_1, q_2, q_3, q_i) \quad (5-9)$$

其中  $q_i$  表示相位  $i$  对应的车道中的排队长度。

### 5.3.4 结合流量预测的 Double DQN 交通灯区域控制算法

5.2 节中对结合流量预测的 Double DQN 算法进行了详细说明，下面将结合流量预测的 Double DQN 算法应用于交通灯区域控制

将路网中的每个交叉口建模为一个 Agent，通过第三章中改进的 Newman 快速算法，将路网划分为一个个较小的控制子区，每个子区内部包含  $n$  个交叉口，一般情况下  $n$  为个位数，对于任意一个交叉口 Agent  $i$ ，它的邻居交叉口为  $\{i_1, i_2, \dots, i_n\}$ ，子区内交叉口可进行信息交换。在  $t$  时刻，交叉口 Agent  $i$  拥有自身状态和动作的历史数据  $\{(s_1, a_1), (s_2, a_2), \dots, (s_{t-1}, a_{t-1})\}$ 。

在各个子区内部，按照本文 5.2 节中的方法，在每个时间步  $t$  中，交叉口 Agent  $i$  与环境进行交互，得到当前状态  $s_t$ 。与此同时，Agent  $i$  将当前状态发送到交通流预测模型之中。交通流预测模型依据当前的交通流状态  $s_t$  对下一时刻的交通流状态进行预测，得到下一时刻的预测状态  $s_{t+1}'$ ，并将预测结果  $s_{t+1}'$  返回给 Agent  $i$ 。Agent  $i$  计算当前状态和预测将来状态中的最优  $Q$  值，并进行动作的选择，然后根据状态和动作获得对应的奖励值，最后对  $Q$  值进行更新。通过训练每个 Agent 以找到局部最优  $Q$  值，然后交互每个 Agent 的局部最优  $Q$  值的信息，将  $Q$  值转移到当前值网络计算误差函数，然后使用梯度下降算法来更新网络参数  $\theta$ 。完成对全局最优  $Q$  值的寻找。因为不仅考虑各交叉口自身的  $Q$  值，而且考虑邻居交叉口的  $Q$  值，所以可以实现交通信号灯区域的协同控制。最后，可以得到每个 Agent 的全局最优  $Q$  值。Agent  $i$  的  $Q$  值按以下公式 (5-10) 更新：

$$Q_i^{t+1}(s_i^t, a_i^t) = (1-\alpha)Q_i^t(s_i^t, a_i^t; \theta) + \alpha * (R^{t+1} + \gamma \max_{a'} Q_i^t(s_i^{t+1}, a'; \theta^-)) \\ + \sum_{j \in N_{nei}} r_{ij} Q_j^{t-1}(s_i^{t-1}, a_i^{t-1}; \theta) + \sum_{j \in N_{oth}} c Q_j^{t-1}(s_i^{t-1}, a_i^{t-1}; \theta) \quad (5-10)$$

其中  $N_{nei}$  是交叉口 Agent i 的相邻交叉口集合, 其中  $N_{oth}$  是交叉口 Agent i 所在子区内的其他交叉口集合,  $s_i^{t+1}$  为交通流量预测模型在时间步  $t+1$  预测的交通流状态,  $a'$  是  $Q_i^t(s_i^{t+1}, a)$  中使 Q 值最大的动作,  $\theta$  代表评估网络的参数,  $\theta^-$  代表目标网络的参数,  $r_{ij}$  为交叉口 Agent i 和 Agent j 的关联系数,  $c$  是折扣系数,  $Q_i^t(s_i^t, a_i^t)$  为在时间步  $t$  时, Agent i 在状态  $s_i$  选取动作  $a_i$  时的 Q 值。在更新 Q 值时使用交通流量预测模型的预测值代替  $s_{t+1}$ 。通过流量预测, 新更新的 Q 值将更加接近真实数据。

在路网的任意一个子区域中, 结合流量预测的 Double DQN 交通灯区域控制方法的学习过程如下:

步骤 1: 对路网进行子区划分, 对各个子区中交叉口的信号控制器 Agent i 进行初始化, 包括状态矩阵, 折扣因子  $\gamma$ , 评估网络参数  $\theta$  和目标网络参数  $\theta^-$ ,  $\varepsilon$ -greedy 中的探索率  $\varepsilon$ , 即时奖励  $R$ , 目标网络更新间隔  $C$ , 训练时间步  $t$ , 训练轮次  $T$ , 当前训练步  $t=1$ 。

步骤 2: 获得当前状态  $s_t^i$ , 将  $s_t^i$  输入到评估网络中, Agent i 得到评估网络中所有动作对应的 Q 值输出。同时将  $s_t^i$  发送给流量预测模块, 根据公式 (5-3) 得到下一时刻的预测状态  $s_{t+1}^i$ , 并将预测结果  $s_{t+1}^i$  返回给 Agent i。Agent i 计算前状态和预测将来状态中的最优 Q 值, 选择并执行动作  $a$ , 获得奖励值  $R$ , 获得下一状态  $s_{t+1}$ 。

步骤 3: 将经验  $experience = (s_t, a, R, s_{t+1}, s_{t+1}^i)$  存储在回放经验池  $D$  中。更新状态  $s_t \leftarrow s_{t+1}$ 。

步骤 4: 从回放经验池  $D$  中进行数据采样, 结果输入到网络中, 根据公式 (5-5) 计算当前路口 Q 值。获取邻居路口的 Q 值, 根据公式 (5-10) 更新 Q 值。

步骤 5: 更新评估网络参数  $\theta$  和目标网络参数  $\theta^-$ , 并对所有路口 Agents 执行上述步骤。

步骤 6: 若  $t < T$  并且  $s_t^i$  不是终止状态, 转步骤 2。

## 5.4 实验设计及结果分析

### 5.4.1 实验设计

为了验证本文提出的 TC-TFP 算法的性能, 本文将对 TC1 算法、TCLJ 算法、MaxplusJJ 算法、MaxplusLJ 算法和 TC-TFP 算法在 GLD 仿真平台中进行实验和仿真。以上五种算法将在图 3.1 所示的路网中进行仿真实验, 对于轻度交通流的环境以及重度交通流的环境分别进行五次实验, 计算各评价指标五次实验结果的平均值。

车辆产生频率的设置: 在文献<sup>[61]</sup>中说明车辆产生频率值是 0.4 时属于高度交通需

求阈值，当车辆产生频率超过 0.4 时，任何交通信号控制方法都无法避免该网络发生拥堵。实验中包含 Bus 和 Car 两种车辆。在轻度交通流下，每个边缘交叉口的 Car 产生频率设为 0.2，即 1 分钟 12 辆，Bus 的产生频率设为 0.05，即 1 分钟 3 辆；在重度交通流下，每个边缘交叉口的 Car 的产生频率设为 0.3，即 1 分钟 18 辆，Bus 的产生频率设为 0.05，即 1 分钟 3 辆。

对比算法参数设置：在 TC1 算法、TCLJ 算法、MaxplusJJ 算法、MaxplusLJ 算法中，设置折扣因子  $\gamma=0.9$ ， $\varepsilon$ -greedy 策略中的  $\varepsilon=0.01$ 。

TC-TFP 算法中的神经网络包 1 个输入层，3 个全连接层和 1 个输出层。输入层神经元个数为 8，全连接层层神经元个数为 40，40，40，输出层神经元个数为 4。

TC-TFP 算法中的参数设置如表 5.3 所示：

表 5.3 TC-TFP 算法参数设置

参数	值
批度大小	32
学习率	0.005
折扣率	0.99
经验回放单元容量	30000
目标网络更新步长	3000
$\varepsilon$ -greedy 中的 $\varepsilon$	由 1 递减至 0.01
迭代次数	50000

## 5.4.2 结果分析

### (1) 轻度交通流环境

图 5.5，图 5.6 分别为轻度交通流环境下 5 种算法的 ATWT 和 AJWT 的图像。表 5.4 为轻度交通流环境下各种算法 5 次仿真试验平均评价指标。

可以看出在轻度交通流环境下，虽然 5 种算法都可以避免交通拥堵的发生，但是每种评价标准上，TC-TFP 算法比其他算法表现更加良好，从 ATWT 和 AJWT 图像能够发现 TC-TFP 算法波动较小，更加稳定，并且能够快速减少等待时间并保持相对稳定。TC-TFP 算法能够解决轻度交通流环境下的交通拥堵问题。



表 5.4 轻度交通流环境下各种算法 5 次仿真试验平均评价指标

评价参数 方法	ATWT (cycle)	AJWT (cycle)	TVA (辆)	实际仿真 时间	预期仿真 时间
TC1	4.6542	1.4573	169378	50000	50000
TCLJ	4.6081	1.4496	169402	50000	50000
MaxplusJJ	5.5776	1.7648	168937	50000	50000
MaxplusLJ	4.5859	1.4295	169296	50000	50000
TC-TFP	4.3984	1.3062	170635	50000	50000

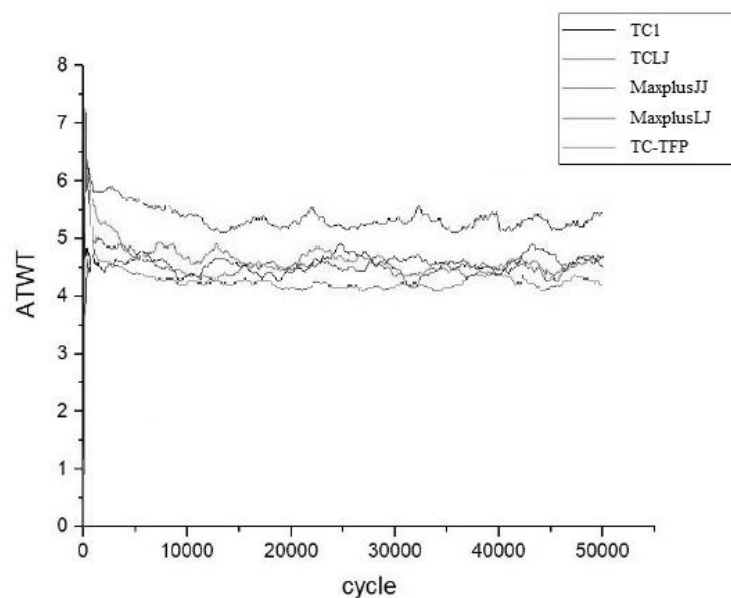


图 5.5 轻度交通流环境下 5 种算法的 ATWT 图像

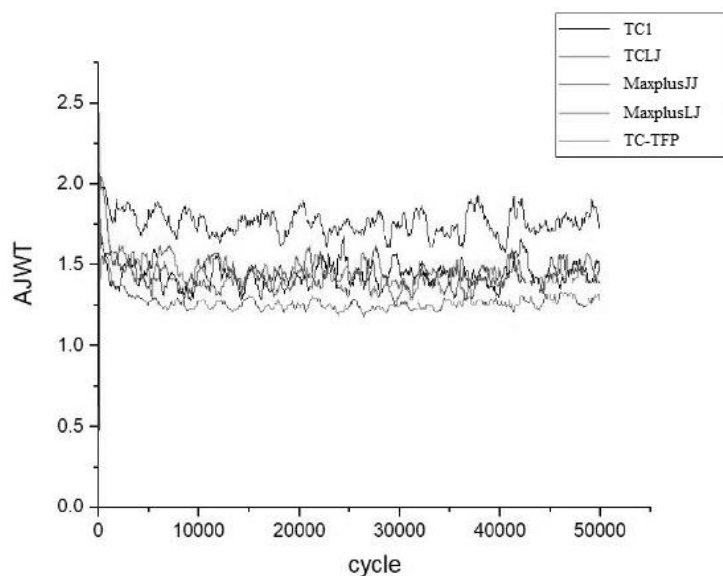


图 5.6 轻度交通流环境下 5 种算法的 AJWT 的图像

## (2) 重度交通流环境

图 5.7, 图 5.8 分别为重度交通流环境下 5 种算法的 ATWT 和 AJWT 的图像。表 5.5 为重度交通流环境下各种算法 5 次仿真试验平均评价指标。

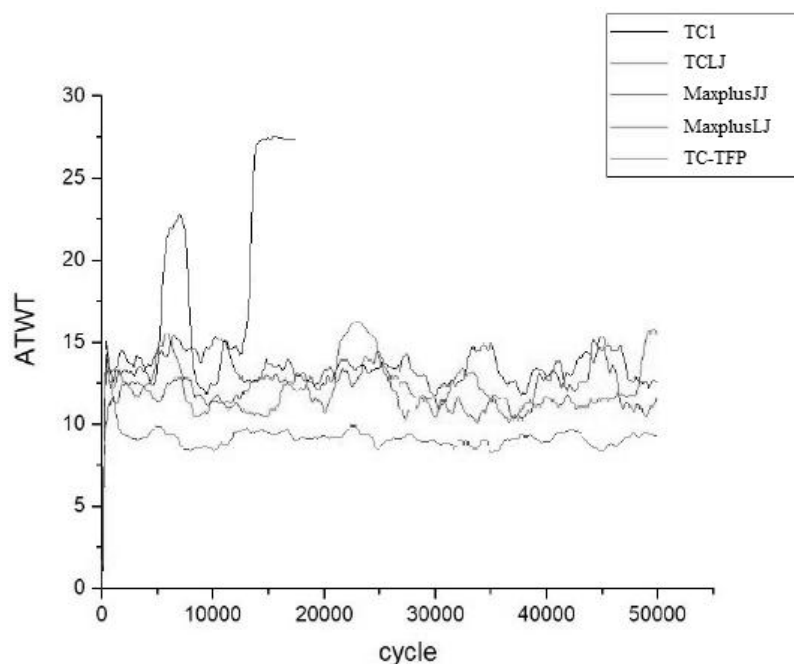


图 5.7 重度交通流环境下 5 种方法的 ATWT 图像

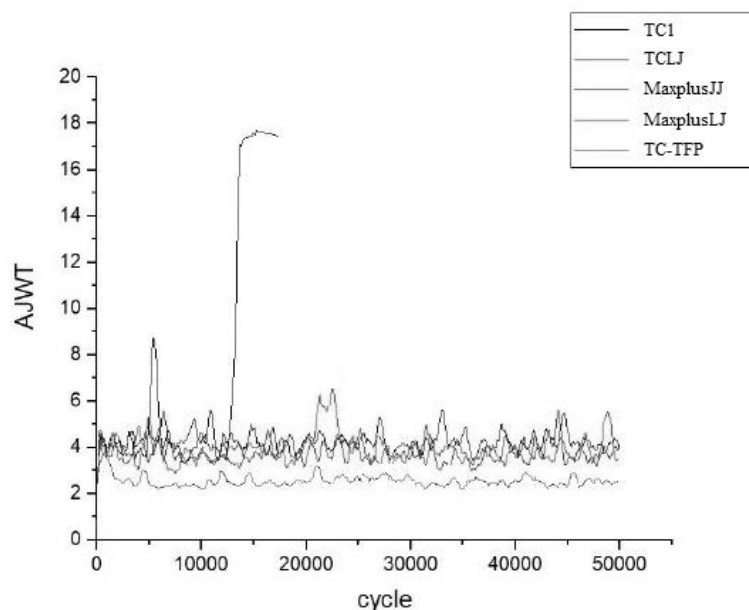


图 5.8 重度交通流环境下 5 种方法的 AJWT 的图像

表 5.5 重度交通流环境下各种算法 5 次仿真试验平均评价指标

评价参数 方法	ATWT (cycle)	AJWT (cycle)	TVA (辆)	实际仿真 时间	预期仿真 时间
TC1	18.1933	7.6952	72558	17532	50000
TCLJ	12.9652	4.2158	218504	50000	50000
MaxplusJJ	14.1024	4.6318	218234	50000	50000
MaxplusLJ	12.3576	4.0256	218935	50000	50000
TC-TFP	9.8974	2.9903	220314	50000	50000

可以看出在重度交通流环境下，TC1 算法发生了交通拥堵，造成了路网瘫痪，仿真提前终止。虽然其它 4 种方法都可以避免交通拥堵的发生，但是每种评价标准上，TC-TFP 算法比其他算法表现更加良好，从 ATWT 和 AJWT 图像能够发现，TC-TFP 算法波动较小、更加稳定，能够解决重度交通流环境下的交通拥堵问题。

综上所述，TC-TFP 算法与 TC1 算法、TCLJ 算法、MaxplusJJ 算法、MaxplusLJ 算法相比，无论在轻度交通流环境下还是重度交通流环境下都能更加有效的解决交通拥堵问题。特别是在重度交通流环境下，TC-TFP 算法能够有效控制路网状况，提高车辆的通行能力。TC-TFP 算法通过结合子区划分和流量预测，能够对未来的交通状况进行预判，使控制策略更具有前瞻性，保障了路网平稳运行。因此，TC-TFP 算法

可以通过控制交通信号灯来降低车辆的等待时间，提高车辆的通行效率，保持路网畅通。

## 5.5 本章小结

本章提出了一种基于深度强化学习的交通灯区域控制方法。在 Double DQN 的基础上，通过结合子区划分和流量预测，实现多交叉口协同控制。结合流量预测的 Double DQN 交通灯区域控制方法解决了现有的信号控制算法中没有很好解决多交叉口协同控制，且控制策略具有一定的滞后性的问题。实验表明，TC-TFP 算法在轻度交通流环境和重度交通流环境下都能够有效降低车辆的等待时间，提高了车辆的通行效率，且控制效果明显优于其他方法。

## 第六章 总结与展望

### 6.1 工作总结

近年来,随着中国现代化建设的稳步前进,城市化进程的不断加快,机动车的数量也大幅增加,交通拥堵已经成为各大城市普遍存在的问题。仅仅依靠扩建道路基础设施已经不能解决城市交通的供需矛盾,而是应当通过优化交通信号控制方法实现对道路交通的有效控制。由于交通流特性复杂,路网规模庞大,将路网划分为若干个相互独立的子区进行控制是必然的发展结果。而准确的交通预测模型能够对接下来的交通状况更好的进行预判,结合当下和未来的交通情况,制定合理的交通信号控制策略,对路网的控制具有重要影响。所以通过有效划分子区,提高对交通流的预测能力,建立合适的交通灯控制模型,有助于解决交通拥堵问题。

本文主要对基于深度强化学习的交通灯区域控制方法在交通信号控制中的应用做了相关探索,主要工作总结如下:

1、对传统 Newman 快速算法进行改进,综合考虑了道路交通流量、路段长度、排队长度、信号周期、交通流密度等特征,定量分析交叉口间关联度,改进模块度  $Q$  的计算方法,实现对路网的动态划分。通过仿真实验进行对比,证明了基于改进的 Newman 快速算法的动态控制子区划分效果优于传统 Newman 快速算法。

2、对传统交通流预测算法进行改进,综合考虑了考虑交通流的空间特征和时间特征,使用 CNN 和 GRU 网络对交通流的空间特征和时间特征进行提取。通过仿真实验进行对比,证明了基于 CNN 和 GRU 网络的交通流预测算法优于传统 LSTM 预测算法。

3、对 Double DQN 算法进行改进,结合子区划分算法和流量预测算法,设计并实现了一种基于深度强化学习的交通灯区域控制算法。通过仿真实验进行对比,证明了在轻度交通流和重度交通流下,基于深度强化学习的交通灯区域控制算法都优于传统的交通灯控制算法。

### 6.2 研究展望

本文针对子区划分、流量预测、交通灯区域控制方法三方面进行了改进和融合,取得了较好的性能。但是也存在一些不足,在未来的研究工作中可以从以下几个方面进一步深入探讨:

1、在本文的研究中,路网中各交叉口的信号控制系统在进行决策时,需要用到路网子区域内其它交叉口的信息。如何高效完成信息的交换,提高信息的传递效率是需要解决的问题。

2、在本文的研究中，使用的交通流预测算法还有许多影响因素没有考虑，如天气、节假日等。如何根据众多的影响因素制定更加准确的交通流预测算法是需要解决的问题。

3、在本文的研究中，仅考虑了几种机动车辆对交通信号控制方法的影响。现实中道路环境更加复杂，不仅机动车种类繁多，非机动车和行人对交通信号控制方法也具有很大的影响。如何在交通信号控制算法中引入非机动车和行人的影响是需要解决的问题。

## 参考文献

- [1] WALINCHUS R J. Real-time network decomposition and subnetwork interfacing [J]. Highway Research Record, 1971, 366: 20 – 28.
- [2] YAGODA N H, PRINCIPE E H, VICK C E, et al. Subdivision of signal systems into control areas [J]. Traffic Engineering, 1973, 43(12): 42 – 45.
- [3] PINNELL C, WILSHIRE M R L. Area wide multilevel traffic control systems [J]. IFAC Proceedings Volumes, 1976, 9(4): 339 – 348.
- [4] MOORE J E, JOVANIS P P. Statistical designation of traffic control subareas [J]. Journal of Transportation Engineering, 1985, 111(3): 208 – 223.
- [5] CHANG E C P. How to decide the interconnection of isolated traffic signals[C]//Proceedings of the 17th conference on winter simulation. New York: ACM, 1985, 445—453
- [6] 李瑞敏, 陆化普, 史其信. 交通信号控制子区模糊动态划分方法研究[J]. 武汉理工大学学报: 交通科学与工程版, 2008, 32(3): 381-384.
- [7] 徐建闽, 鄢小文, 荆彬彬, 等. 考虑交叉口不同饱和度的路网动态分区方法[J]. 交通运输系统工程与信息, 2017, 17(4): 149 – 156.
- [8] Hu Y, Wang Y, Zhang J, et al. Correlation degree analysis of arterial adjacent intersections for coordinated control subunit partition[J]. Advances in Mechanical Engineering, 2018, 10(1): 168781401774874.
- [9] 刘林. 基于 LSSVM 的短期交通流预测研究与应用[D]. 成都: 西南交通大学, 2011.
- [10] SMITH B L, DEMETSKY M J. Traffic Flow Forecasting: Comparison of Modeling Approaches[J]. Journal of Transportation Engineering, 1997, 123(4): 261-266.
- [11] Messer C, Urbanik II. Short-Term Freeway Traffic Volume Forecasting Using Radial Basis Function Neural Network[J]. Transportation Research Record Journal of the Transportation Research Board, 1998(1651): 39-47.
- [12] Kumar K, Parida M, Katiyar V K. Short term traffic flow prediction for a non urban highway using artificial neural network[J]. Procedia-Social and Behavioral Sciences, 2013, 104: 755-764.
- [13] Lv Y, Duan Y, Kang W, et al. Traffic Flow Prediction With Big Data: A Deep Learning Approach[J]. Intelligent Transportation Systems, IEEE Transactions on, 2015, 16(2): 865-873.
- [14] Ma X, Tao Z, Wang Y, et al. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data[J]. Transportation Research Part C Emerging Technologies, 2015, 54: 187-197.
- [15] Xiaolei M, Zhuang D, Zhengbing H, et al. Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction[J]. Sensors, 2017,

- 17(4).
- [16] Polson N G, Sokolov V O. Deep learning for short-term traffic flow prediction[J]. Transportation Research Part C Emerging Technologies, 2017, 79:1-17.
- [17] Zhang J, Zheng Y, Qi D, et al. DNN-based prediction model for spatio-temporal data[C]. ACM Sigspatial International Conference on Advances in Geographic Information Systems.ACM,2016:92.
- [18] Zhang J, Zheng Y, Qi D, et al. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction[J]. National Conference on Artificial Intelligence,2016:1655-1661.
- [19] Yuankai Wu, Huachun Tan, Lingqiao Qin et al A hybrid deep learning based traffic flow prediction method and its understanding[J]. Transportation Research Part C: Emerging Technologies Volume 90,May 2018, Pages 166-180
- [20] 邓焜堃, 万 良, 丁红卫. 基于深度学习的交通流量预测研究[J]. 计算机工程与应用, 2019, 55(2): 228-235.
- [21] 罗向龙,焦琴琴,牛力瑶等.基于深度学习的短时交通流预测 [J]. 计算机应用研究,2017,34(1):91-93.
- [22] Kuyer L, Bakker B, Whiteson S. Multiagent reinforcement learning and coordination for urban traffic control using coordination graphs and max-plus[D]. MS Thesis, 2007.
- [23] Kuyer L, Whiteson S, Bakker B, et al. Multiagent reinforcement learning for urban traffic control using coordination graphs[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, Heidelberg, 2008: 656-671.
- [24] Li L, Lv Y, Wang F Y. Traffic signal timing via deep reinforcement learning[J]. IEEE/CAA Journal of Automatica Sinica, 2016, 3(3): 247-254.
- [25] Rosyadi A R, Wirayuda T A B, Al-Faraby S. Intelligent traffic light control using collaborative Q-Learning algorithms[C]//2016 4th International Conference on Information and Communication Technology (ICo ICT). IEEE, 2016: 1-6.
- [26] Vidhate D A, Kulkarni P. Cooperative multi-agent reinforcement learning models (CMRLM) for intelligent traffic control[C]//2017 1st International Conference on Intelligent Systems and Information Management (ICISIM). IEEE, 2017: 325-331.
- [27] Surya S, Rakesh N. Flow based traffic congestion prediction and intelligent signalling using Markov decision process[C]//2016 International Conference on Inventive Computation Technologies (ICICT). IEEE, 2016.
- [28] 伦立宝. 基于强化学习的城市交通信号控制方法研究[D]. 西安电子科技大学, 2013.
- [29] 夏新海. 面向城市自适应交通信号控制的强化学习方法研究[D]. 广州: 华南理工大学,2013.



- [30] 王莹多. 基于深度强化学习的路口自适应控制[D].大连理工大学, 2017.
- [31] 闫呈祥. 基于深度强化学习的城市交通灯控制方法研究[D]. 西安电子科技大学, 2019.
- [32] Lin Y Q, Li M, Chen X C, et al. A Belief Rule Base Approach for Smart Traffic Lights[C]// International Symposium on Computational Intelligence & Design. IEEE, 2016.
- [33] Chen Y R, Chen K P, Hsiung P A. Dynamic traffic light optimization and Control System using model-predictive control method[C]//IEEE International Conference on Intelligent Transportation Systems. IEEE, 2016.
- [34] 马旭辉, 滑亚飞, 何忠贺. 基于路网可达性的交通控制子区划分方法研究[J]. 计算机工程与应用, 2016, 52(8): 1-6.
- [35] Gong Yishan, Zhang Yi. Research of Short-Term Traffic Volume Prediction Based on Kalman Filtering[C]// 2013 6th International Conference on Intelligent Networks and Intelligent Systems (ICINIS). Shenyang, China: IEEE Press, 2013: 99-102.
- [36] Xu Dan, Shi Yijie. A combined model of random forest and multilayer perceptron to forecast expressway traffic flow[C]// 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC). Macau: IEEE Press, 2017: 448-451.
- [37] Xu Jie, Deng Dingxiong, Demiryurek U, et al. Mining the Situation: Spatiotemporal Traffic Prediction With Big Data[J]. IEEE Journal of Signal Topics in Signal Processing, 2015, 9(4): 702-715.
- [38] Wang J, Hu F, Li L. Deep Bi-directional Long Short-Term Memory Model for Short-Term Traffic Flow Prediction[J]. 2017.
- [39] Cheng X, Zhang R, Zhou J, et al. DeepTransport: Learning Spatial-Temporal Dependency for Traffic Condition Forecasting[J]. 2017.
- [40] Hubel D H, Wiesel T N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex[J]. The Physiological Society, 1962, 160(1): 106-154.
- [41] Kuniyiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological Cybernetics, 1980, 36(4): 193-202.
- [42] Le Cun Y, Bottou L, Bengio Y, and Haffner P. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
- [43] Krizhevsky A, Sutskever I, Hinton G. ImageNet Classification with Deep Convolutional Neural Networks[C]// NIPS. Curran Associates Inc. 2012.
- [44] Szegedy C, Liu W, Jia Y, et al. Going Deeper with Convolutions[J]. 2014.
- [45] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE Computer Society, 2016.

- 
- [46] Hochreiter S, Schmidhuber J. Long Short-term memory[J]. Neural Computation, 1997,9(8):1735-1780.
  - [47] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[J]. Computer Science, 2014.
  - [48] H Leung, Faouzi N E E, Kurian A. Intelligent Transportation System(ITS)[J]. Information Fusion, 2011,12(1):2-3.
  - [49] Guo L, Yang R, Zhang M. Arterial Traffic Two-Direction Green Wave Coordination Control Based on MATLAB Graphical Method[C]// International Conference on Information Science and Control Engineering. IEEE, 2015.
  - [50] 高雨, 沈国江, 叶炜. 城市路网交通分散协调控制[J]. 信息与控制, 2005,34(5):616-620.
  - [51] Chen Z M, Liu X M, Wu W X. Optimization Method of Intersection Signal Coordinated Control Based on Vehicle Actuated Model[J]. Mathematical Problems in Engineering, 2015,3(1):1-9.
  - [52] Kernighan B W, Lin S. An efficient heuristic procedure for partitioning graphs[J]. Bell System Technical Journal, 1970,49(2):291-307.
  - [53] 汪小帆, 李翔, 陈关荣. 复杂网络理论及其应用[M]. 清华大学出版社, 2006.
  - [54] Dan S, Fusco J, Schank P, et al. Discovery of Community Structures in a Heterogeneous Professional Online Network[C]// Hawaii International Conference on System Sciences. IEEE, 2013:3262-3271.
  - [55] GIRVAN M, NEWMAN M E J. Community structure in social and biological networks [J]. Proceedings of the National Academy of Sciences of the United States of America, 2002, 99(12): 7821 - 7826.
  - [56] NEWMAN M E J, GIRVAN M. Finding and evaluating community structure in networks [J]. Physical Review E, 2004, 69(2): 026113.
  - [57] NEWMAN M E J. Fast algorithm for detecting community structure in networks [J]. Physical Review E, 2003, 69(6): 066133.
  - [58] Mnih V, Kavukcuoglu K, Silver D, et al. Playing Atari with Deep Reinforcement Learning[J]. Computer ence, 2013.
  - [59] Van Hasselt Hado, Guez Arthur, Silver David. Deep Reinforcement Learning with Double Q-learning[C]//AAAI.2016.16:2094-2100.
  - [60] Wang Z, Schaul T, Hessel M, et al. Dueling Network Architectures for Deep Reinforcement Learning[J]. arXiv Prepr arXiv:1511.06581. 2015.
  - [61] Marco Wiering, Jelle van Veenen, Jilles Vreeken et al. Intelligent Traffic Light Control[J]. Technical report, Department of Information and Computing Sciences, University Utrecht. 2004.1-30.

## 致谢

光阴似箭，日月如梭，三年的研究生生活转瞬即逝。在这三年当中，我遇到许多有趣的人、有趣的事，也从中学到了许多知识。在此，衷心地向所有给予我鼓励和帮助的人献上诚挚的祝福！

首先，感谢我的导师方敏教授，她为人正直、善良、和蔼可亲，在三年的研究生生活中，她悉心教导、耐心督促并给予我很多鼓励。方老师在研究课题的选择和创新点的优化提取上，给予我耐心的指导。同时，方老师严谨的治学态度、敏锐的洞察力和渊博的专业知识、精益求精的工作作风也深深影响着我，让我受益终身！正是方老师的悉心教导让我顺利完成论文，在此谨向方老师致以诚挚的谢意和崇高的敬意！

其次，感谢实验室的小伙伴们，感谢你们陪我度过三年的美好时光，在学习和生活上对我的支持和帮助以及在科研上提出的宝贵建议。在此向实验室的师兄师姐师弟师妹们表示由衷的感谢！

同时，感谢家人在读研期间对我的关照和鼓励，他们是我坚强的后盾，让我能够专心的投入学业之中，在学业完成之际向我的家人表示深深的感谢！

最后，感谢参与我论文评审和答辩的各位老师，感谢你们在百忙之中评阅我的论文，你们提出的宝贵意见和诚恳批评将使我受益匪浅。

再次感谢出现在我生活中的所有人，在今后的生活中我会更加努力，不负老师同学家人们的期望。同时，把最好的祝福献给你们，愿你们一生幸福安康！



## 作者简介

### 1. 基本情况

丁献硕，男，山东聊城人，1994 年 5 月出生，西安电子科技大学计算机学院计算机科学与技术专业 2017 级硕士研究生。

### 2. 教育背景

2013.08~2017.07 西安电子科技大学，本科，专业：计算机科学与技术

2017.08~            西安电子科技大学，硕士研究生，专业：计算机科学与技术

### 3. 攻读硕士学位期间的研究成果

[1] 参与部委研究项目一项，2018-2019，已交付，主要负责人。