

基于SUMO的虚拟交通灯的设计与实现

赵会奋¹, 蔡文钦¹, 宋 玮¹, 蒋 毅²

¹广东工业大学计算机学院, 广东 广州

²苏州易维迅信息科技有限公司, 江苏 苏州

收稿日期: 2022年4月13日; 录用日期: 2022年5月10日; 发布日期: 2022年5月18日

摘 要

现行的传统物理交通信号灯存在着不能很好适应复杂的交通情况和高额的维护费用等弊端, 一种解决方法是采取虚拟交通灯方案, 依托车联网技术, 通过车车交互来选择领导车辆, 并由领导车辆承担传统交通灯的工作, 负责绿灯通行权的授予和回收。本文基于SUMO (Simulation of Urban Mobility)搭建了虚拟交通灯平台, 设计了虚拟交通灯平台的车辆状态、消息格式以及核心控制算法。通过在平台上实现传统的交通灯与虚拟交通灯进行比较实验, 证明虚拟交通灯相较于传统物理交通灯模式性能上有很大的提升。本文搭建的基于SUMO的虚拟交通灯平台可以为今后的深入实验和研究提供平台和框架。

关键词

虚拟交通灯, SUMO, 领导车辆选举, 绿灯授予和回收

Design and Implementation of Virtual Traffic Light Based on SUMO

Huifen Zhao¹, Wenqin Cai¹, Wei Song¹, Yi Jiang²

¹School of Computer, Guangdong University of Technology, Guangzhou Guangdong

²Easyservice Company Limited, Suzhou Jiangsu

Received: Apr. 13th, 2022; accepted: May 10th, 2022; published: May 18th, 2022

Abstract

The traditional physical traffic lights do not work well with the complex traffic situation and need high maintenance costs. One solution is virtual traffic lights which rely on vehicle network. Leader vehicle is selected through the vehicle interaction, and the leader vehicle takes responsibility of traditional traffic lights to grant and revoke green light. This paper builds a virtual traffic lights

platform based on SUMO (Simulation of Urban Mobility), and designs the vehicle status, message format and core control algorithm of the virtual traffic lights. The experiment proves that the performance of virtual traffic lights is greatly improved compared with the traditional traffic lights on the platform. The SUMO-based virtual traffic light built in this paper can provide a platform and framework for future in-depth experiments and research.

Keywords

Virtual Traffic Lights, SUMO, Leader Vehicle Election, Green Light Grant and Revoke

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

近年来,越来越多的城市出现大规模的交通拥堵状况,严重影响着人们的日常出行。缓解交通拥堵的措施一般可分为三种:减少车辆数量(限行措施)、增加基础设施能力(拓宽道路、增加路网密度、优化道路结构等)和增加路口的吞吐量。研究表明,增加路口的吞吐量是最为经济的方式之一,而路口的吞吐量瓶颈与路口的管理存在很大关系。自20世纪初电力交通信号灯开始使用至今,大多数交通灯仍采用固定的时间周期模式,而随着城市的发展,车辆不断增多,而且路况变得愈加复杂,这种固定时间周期的交通信号灯已经不能很好地适用拥挤的路况,这使得智能交通灯成为当前一大研究热点。

所谓智能交通灯,就是根据历史或当前城市交通状态,动态设定路口处交通信号灯相位规则。从而使得路口处车辆可以在尽可能短的时间内有序通过,消除空等,减少平均等待时间。

基于 IEEE802.11p 的车联网技术的不断成熟也促进了智能交通灯的发展,作为智慧城市不可或缺的一部分,一些国家已经开始推广车联网,车联网的普及会使交通信号灯系统发生巨大改变。车联网中车辆可以通过 V2I 和 V2V 的形式进行通信,以此来共享车辆实时信息。

虚拟交通灯(Virtual Traffic Lights, VTL)的提出开启了智能交通灯的另一个方向,即在车联网中采用虚拟交通灯来解决交通拥堵问题,该方法不需要任何街边基础设施,而是让车辆间进行直接通信,协商出路权分配规则,并通过车载显示装置来告知司机交通信号灯,可以极大节省修建交通信号灯的费用,并且使得交通信号灯无处不在。

本文基于 SUMO (Simulation of Urban Mobility)搭建了虚拟交通灯平台,设计了虚拟交通灯平台的车辆状态、消息格式以及核心控制算法等。通过与传统的交通灯进行对比,证明虚拟交通灯相较于传统物理交通灯性能有很大的提升,为今后的深入研究提供平台和框架。

2. 国内外现状

文献[1]首次提出这个概念,虚拟交通灯包括三个部分:首先,领导者选举,通过算法选举出领导车辆来代替实体交通灯;其次,交通控制,领导车辆分配路权规则并发送给车辆。最后是领导权转移,领导车辆离开路口前进行领导权转移,由下一辆获得领导权的车辆继续控制路口。可是文中未给出详细的算法过程。文献[2]则给出了详细交互过程并改进了[1]中的车辆消息传播方式。文献[3]提出为绿灯通行车辆设定阈值来控制通行时间,文献[4]在[3]的基础上,提出独立路口虚拟交通灯相位算法,通过各车道车辆的等待时间,排队长度以及排队数量这三个参数算出优先级,从而减少等待时间。文献[5]中提出了能

让通行方向不冲突的车辆同时通行的算法，从而增加吞吐量，减少平均行驶时间。除了改进交通控制逻辑的方法，还有的研究分别专注于虚拟交通灯与车载用户界面结合后对主驾驶安全性的影响[6]；领导车辆算法的改进，如选举问题中一致性达成问题[7]和减少交互通信量问题[8]。

本文基于 SUMO 搭建了一个虚拟交通灯平台，设计了相关的车辆状态、车辆间通信的消息格式和消息类型，并实现了简单的虚拟交通灯控制算法，包括领导车辆的选举，绿灯通行权的授予与回收，领导车辆的传递。对比实验表明虚拟交通灯比传统物理交通灯性能有很大的提升。

3. 虚拟交通灯的设计

本部分主要介绍虚拟交通灯的总体设计。不同状态下车辆的行为应该有所不同，因此，在 3.1 给出了车辆的三种状态及其描述。其次，为了实现车辆间的信息交互，在 3.2 和 3.3 中分别给出了车辆间通信的消息格式和消息类型。接下来的 3.4、3.5 和 3.6 则介绍了控制算法的核心内容，分别为领导车辆的选举、绿灯通行权的授予与回收、领导车辆的传递。

3.1. 车辆状态描述

交通灯的作用是实现路口资源的分配，根据路口资源的使用情况为车辆设置了三种状态，分别为 FREE 状态、DANGEROUS 状态和 MOVING 状态。

FREE 状态：该状态表示当前没有其他道路上的车辆与该车辆抢占路口资源。处于 FREE 状态的车辆可以不加任何控制地通过当前所处路口。

DANGEROUS 状态：该状态表示当前有其他道路上的车辆与该车辆抢占路口资源。处于 DANGEROUS 状态的车辆需要获得路口的通行权才能通过当前路口，如果不对该状态车辆加以控制，会有发生碰撞的危险。

MOVING 状态：该状态表示当前车辆已经获得了路口资源的使用权，处于该状态下的车辆可以通过当前路口。

车辆状态转换图如图 1 所示：

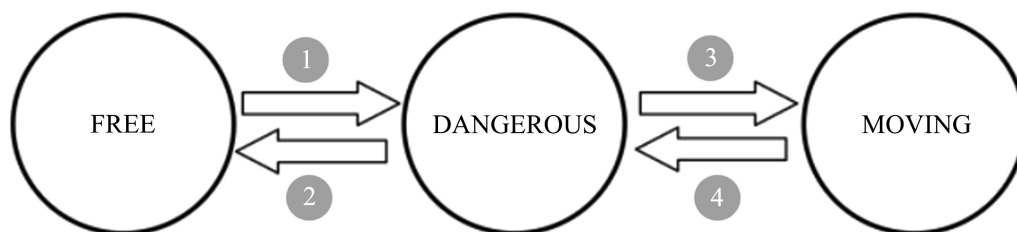


Figure 1. Diagram of the vehicle status transition

图 1. 车辆状态转换图

①：当存在通行车辆的道路数大于 1 时，车辆从 FREE 状态转化成 DANGEROUS 状态。

②：当其他道路车辆离开路口时，剩下一条道路上的车辆将从 DANGEROUS 状态转化成 FREE 状态。

③：当处于 DANGEROUS 状态的车辆获得 leader 车辆授予的绿灯通行权时，将会从 DANGEROUS 状态转化成 MOVING 状态。

④：当处于 MOVING 状态的车辆被 leader 收回绿灯通行权时，将会从 MOVING 状态转化成 DANGEROUS 状态。

3.2. 消息格式

车辆间的通信按照固定的消息格式，本文基于简化原则，为车辆设置的消息格式如表 1 所示：

message (source, destination, message_type, intersection, content)

Table 1. Message format

表 1. 消息格式

消息字段	意义
source	消息的发送方，其值为发送消息的车辆 ID
destination	消息的接收方，其值为接受消息的车辆 ID
message_type	消息所属的类型，其值为表 2 所示
intersection	发送消息的车辆当前所处的路口标志
content	消息所携带的内容

3.3. 消息类型

车辆间通信所使用的消息类型如表 2 所示。

Table 2. Message type

表 2. 消息类型

消息类型	作用
GREEN_REQUEST	绿灯请求消息：用于向 leader 车辆请求路口绿灯通行权。
GREEN_GRANT	绿灯授予消息：用于 leader 车辆向其他道路车辆授予绿灯通行权。
GREEN_REVOKE	绿灯回收消息：用于 leader 车辆回收其他道路车辆绿灯通行权。
VEHICLE_STATUS	车辆状态消息：用于交换各车辆的即时车辆状态。
LEADER_INQUIRY	leader 询问消息：用于获取当前路口的 leader 车辆。
LEADER_ANSWER	leader 回复消息：用于回复 LEADER_INQUIRY 消息。
LEADER_REQUEST	leader 选举消息：用于发起当前路口的 leader 选举。
LEADER_DELIVER	leader 传递消息：用于 leader 车辆的传递。
LEADER_CLAIM	leader 声明消息：用于 leader 车辆声明自己的身份。
LEADER_CHANGE	leader 改变消息：leader 传递成功后新 leader 发送的声明消息。

3.4. 领导车辆的选举

leader 车辆选举算法是虚拟交通灯方案的主要算法，虚拟交通灯方案需要选举出 leader 车辆来负责每个方向车辆绿灯通行权的授予与回收。如图 2 所示，leader 车辆的选举过程如下：

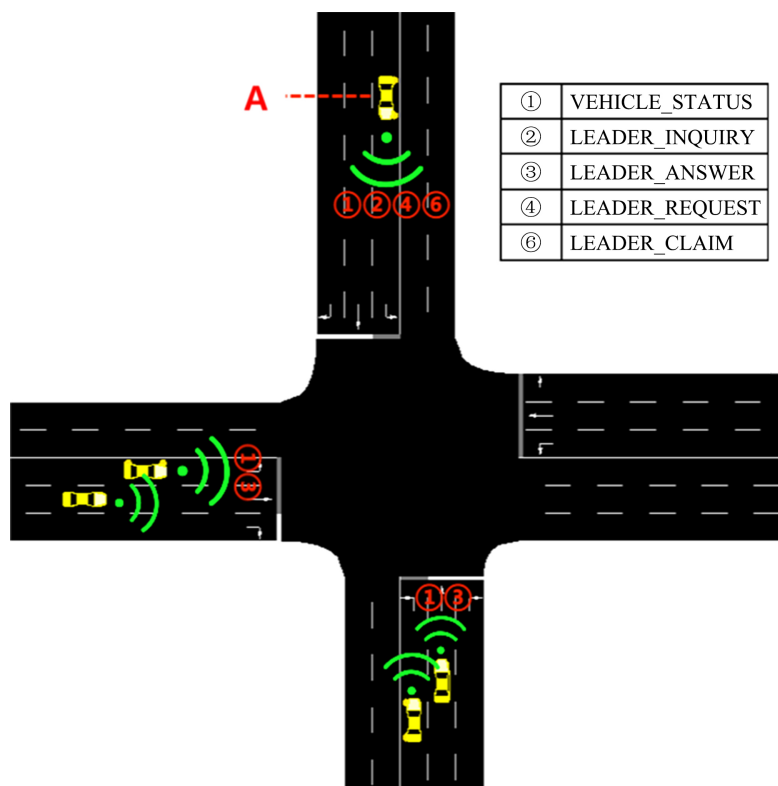


Figure 2. Vehicle election

图 2. 领导车辆选举

① 所有车辆经过固定时间间隔都会广播车辆状态消息(VEHICLE_STATUS), 将自身信息(车辆 ID、所处位置等)广播出去。

② 初始时, A 车辆检查到自身的 currentLeader 属性值为 None, 因此向其他车辆广播 leader 询问消息(LEADER_INQUIRY), 以获取当前路口的 leader 车辆。

③ 其他车辆收到 A 车发过来的 leader 询问消息后, 检查自身 currentLeader 属性, 并通过 leader 回复消息(LEADER_ANSWER)回复车辆 A。由于初始状态时还未存在 leader 车辆, 因此 leader 回复消息中的结果为 None。

④ A 车辆收到其他车辆回复的 leader 回复消息, 判断当前路口还未存在 leader 车辆, 于是广播 leader 选举消息(LEADER_REQUEST), 申请在当前路口进行一次 leader 选举。

⑤ 其他车辆收到 leader 选举消息后, 根据车辆的位置、类型等信息(本算法以车辆到路口的距离作为指标, 距离路口越远的车辆, 优先级越大, 越容易成为 leader), 分布式地计算出当前路口的 leader (图 2 中所示例子选出的 leader 车辆为 A)。

⑥ 当选为 leader 的车辆发出 leader 声明消息(LEADER_CLAIM), 将 leader 车辆身份广播出去

⑦ 其他车辆收到 leader 声明消息后, 将自己 currentleader 值更新。

3.5. 绿灯通行权的授予与回收

若当前路口已经存在 leader 车辆, 其他车辆就需要和 leader 车辆协商通行。具体而言, 就是其他车辆向 leader 车辆申请绿灯通行权, 而 leader 车辆负责绿灯通行权的授予与回收。如图 3 所示(图 3 中所示例子 leader 车辆为 A), 绿灯通行权的授予与回收过程如下:

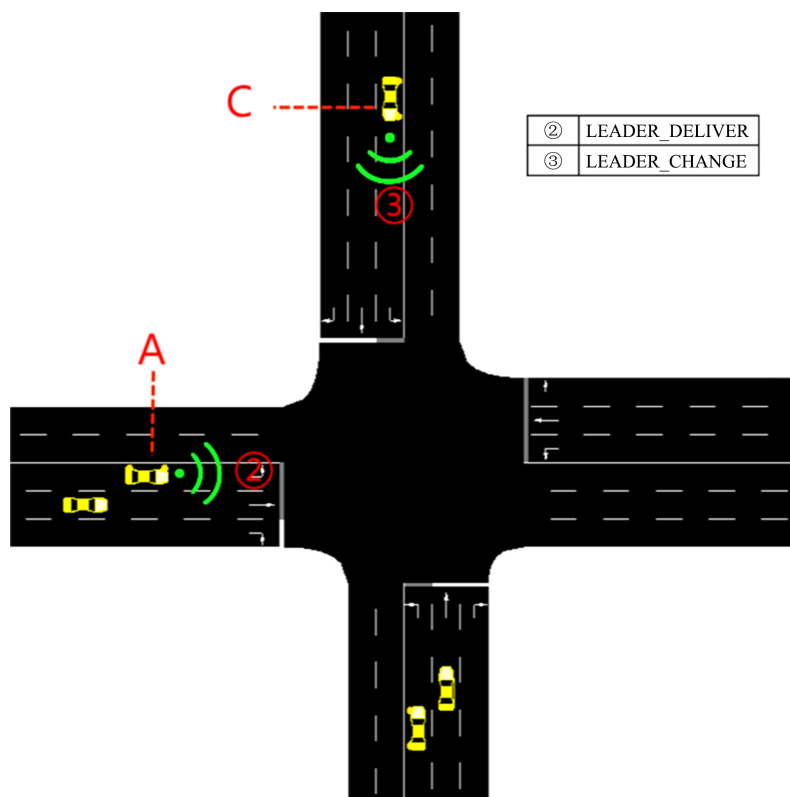


Figure 4. Leader vehicle delivery
图 4. 领导车辆的传递

4. 虚拟交通灯的实现

4.1. SUMO 软件简介

SUMO (Simulation of Urban Mobility)是一款开源、精微的多模式交通仿真软件。该工具无需考虑所有必要的事项即可得到一个完整的交通仿真效果，例如实现或设置用于处理道路网络、需求和交通控制的方法。通过使用通用的体系结构和模型库，使已实现的算法更具有可比性。它能够应对由各种车辆组成的给定交通需求，如何在给定的道路网络中运行进行仿真。该仿真平台可以解决大量的交通管理问题。同时，对于路网中的每辆车，SUMO 都有明确的建模，拥有自己的路线，并能通过网络进行独立地移动。

在 Python 中使用 SUMO 的过程如下：

- 1) 设置环境变量，设置环境变量 SUMO_HOME，值为 SUMO 的安装路径；
- 2) 在 Python 中导入 traci 模块；
- 3) 定义调用的 SUMO 的命令参数列表 sumoCmd：
 - ① sumo-gui 可执行路径，一般为 SUMO_HOME 环境变量拼接“tools”；
 - ② “-c”：说明说明下个参数为配置文件；
 - ③ sumocfg：文件路径，配置了仿真的路网文件和车辆文件；
 - ④ “--tripinfo-output”：说明下个参数为导出文件路径；
 - ⑤ 导出 xml 数据文件路径，仿真过程所有车辆数据的统计数据；
- 4) 调用 traci.start 方法连接，参数为命令参数列表 sumoCmd；
- 5) 编写循环执行 traci.simulationStep 语句。

4.2. 主要类的设计

本文采用 Python 作为搭建虚拟交通灯平台的编程语言。下面表 3 给出主要的类的设计, 包括车辆 (Vehicle)类和车辆控制器类(VehicleController), 这些类的设计具有较好可扩展性, 可作为今后继续研究的平台使用。例如, 本文中按照发送绿灯请求车辆到路口的距离来设置优先级, 如需更换其他排序规则(如按照紧急度排序), 只需更改车辆控制器类的 sortedVehicles 方法即可, 其他部分可以保持不变。

Table 3. Design of major classes

表 3. 主要类的设计

类	属性/方法
Vehicle	vehID: 车辆 ID roadID: 车辆所处道路 ID lanePosition: 车辆当前所处位置 laneLength: 车辆所在车道长度 laneLeader: 车辆所处车辆队列的头车 Intersection: 车辆所处交叉口 vehController: 车辆控制器 setLaneLeader(): 设置车辆的 laneLeader 值 setIntersection(): 设置车辆的 laneLeader 值 setController(): 初始化车辆控制器
VehicleController	Direction: 车辆方向 Leader: 当前路口领导车辆 askVehicle: 存储请求绿灯的车辆 askVehicle_order: 存储排好序请求车辆 Vehicles: 存储当前路口所有车辆信息 status: 车辆状态 Timer: 车辆计时器 priorRoadID: 当前获得通行权的道路 changeStatus(): 根据车辆所处环境切换车辆状态 Run(): 根据车辆状态来决定要进行的动作 Stop(): 让车辆停下来 Resume(): 重置车辆的属性 grantGreen(): 对某一车辆授予绿灯通行权 revokeGreen(): 对某一车辆回收绿灯通行权 sortedVehicles(): 对绿灯请求车辆进行排序 sendMessage(): 发送消息 receiveMessage(): 接收消息

5. 仿真实验及数据

5.1. 实验数据设置

5.1.1. 交叉路口模型

为了简化实验, 本文采用了图 5 所示的双路口道路模型, 但平台也适用于多路口道路模型。

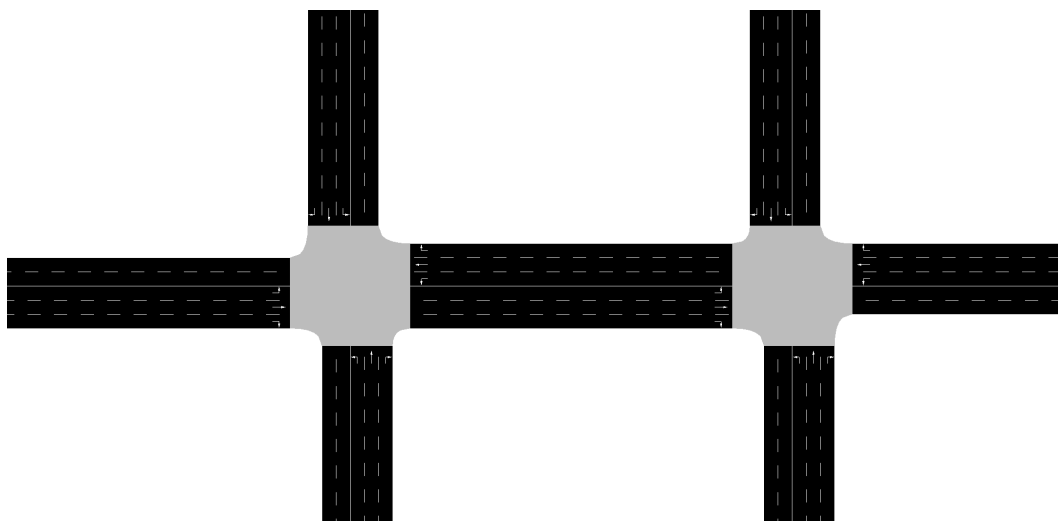


Figure 5. Intersection model
图 5. 交叉路口模型

5.1.2. 车流量设置

图 5 所示的双路口模型共有 6 个方向的出口和 6 个方向的入口, 故共有 $C_6^2 = 15$ 条不同的行驶路线。单条行驶路线车流量定义为每小时每条行驶路线的车辆数。本实验设置单条行驶路线车流量最小值为 10 辆每小时, 最大值为 200 辆每小时, 以 10 为递增量构建 20 组总车流量数据, 设每组数据中每条行驶线路的车流量相同, 则每组总车流量计算为 15 条*单条行驶路线车流量, 如第 1 组总车流量为 15*10 辆每小时, 第 2 组总车流量数据为 15*20 辆每小时, 第 20 组总车流量数据为 15*200 辆每小时。

5.2. 实验结果及分析

实验中传统物理交通灯采用固定时长控制, 绿灯时间固定为 33 秒。实验采用 20 组总车流量数据, 对比两种交通灯下车辆通过图 5 所示路口的总通行时间, 总通行时间越少, 则证明该交通灯性能越好。

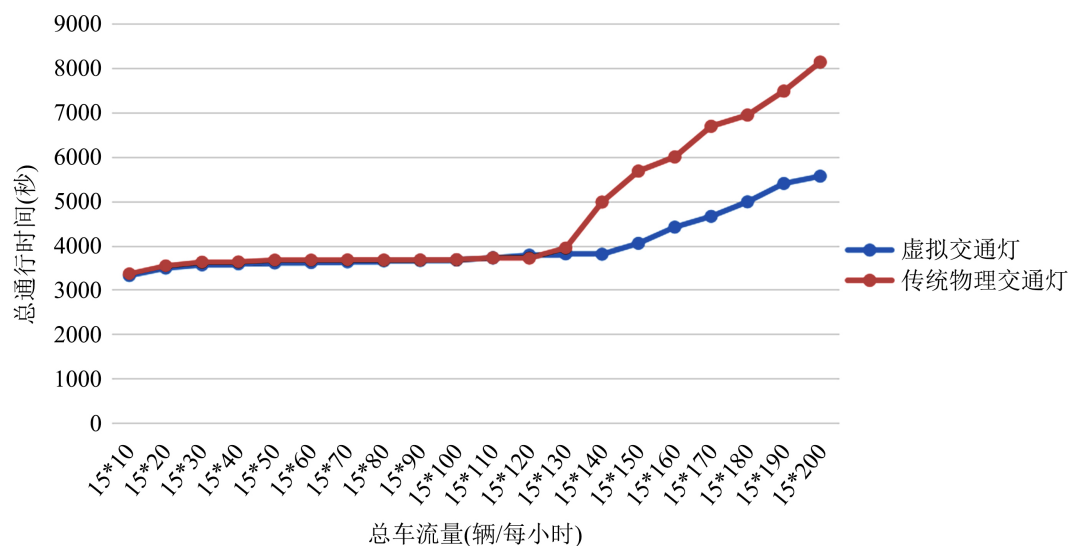


Figure 6. Comparison of experimental results
图 6. 实验结果对比

由图 6 可知,当总车流量小于 15×130 (辆/每小时)时,两种交通灯差距比较小,当总车流量大于 15×130 (辆/每小时)时,两种交通灯在总通行时间上差距越来越大,这证明随着总车流量的增多,虚拟交通灯比传统物理交通灯有更优秀的性能,对路口资源利用率更高。

6. 总结及展望

针对传统物理交通灯时间周期模式固定以及每年需要高额维护费用等弊端,本文设计并实现了基于 SUMO 的虚拟交通灯平台。通行车辆之间通过 V2V 方式进行信息交互、选择领导车辆,并由领导车辆承担传统交通灯的工作,执行绿灯通行权的授予和回收,从而实现绿灯通行权的动态调整。通过在平台上实现传统的交通灯与虚拟交通灯进行比较实验,证明虚拟交通灯相较于传统物理交通灯性能有很大的提升。同时,本文提出的基于 SUMO 的虚拟交通灯平台具有较好的可扩展性,可以适用于其他多路口模型,后续可以将本平台应用于其他虚拟交通灯实验中。

基金项目

广东省省级大学生创新创业训练项目(编号 S202011845159, S202011845160)。

参考文献

- [1] Ferreira, M., Fernandes, R., Conceição, H., et al. (2010) Self-Organized Traffic Control. *Proceedings of the Seventh ACM International Workshop on Vehicular InterNetworking*, Association for Computing Machinery, New York, 85-90. <https://doi.org/10.1145/1860058.1860077>
- [2] Sayin, M.O., Lin, C., Shiraishi, S., et al. (2019) Information-Driven Autonomous Intersection Control via Incentive Compatible Mechanisms. *IEEE Transactions on Intelligent Transportation Systems*, **20**, 912-924. <https://doi.org/10.1109/TITS.2018.2838049>
- [3] Bazzi, A., Zanella, A. and Masini, B.M. (2016) A distributed Virtual Traffic Light Algorithm Exploiting Short Range v2v Communications. *Ad Hoc Networks*, **49**, 42-57. <https://doi.org/10.1016/j.adhoc.2016.06.006>
- [4] 王德. 车联网中虚拟智能交通灯的控制算法研究[D]: [硕士学位论文]. 哈尔滨: 黑龙江大学, 2018.
- [5] Gee, L.M. (2018) Clique-Based Traffic Control Strategy Using Vehicle-to-Vehicle Communication. The University of Western Australia, Perth. https://doi.org/10.1007/978-3-030-29911-8_41
- [6] Monreal, O., Gomes, P., Silveria, M.K., et al. (2012) In-Vehicle Virtual Traffic Lights: A Graphical User Interface. *7th Iberian Conference on Information Systems and Technologies*, Madrid, 20-23 June 2012, 1-6.
- [7] Fathollahnejad, N. (2017) On the Design and Analysis of Consensus Protocols for Vehicular Ad Hoc Network. Chalmers University of Technology, Göteborg, Sweden.
- [8] Zhang, R., Schmutz, F., Gerard, K., et al. (2018) Virtual Traffic Lights: System Design and Implementation. *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Chicago, IL, 27-30 August 2018, 1-5. <https://doi.org/10.1109/VTCFall.2018.8690709>