SONGHANG CHEN, Ph.D.[1]
E-mail: songhang.chen@fjirsm.ac.cn
DAN ZHANG, M.D.[1]
(Corresponding author)
E-mail: zdan@fjirsm.ac.cn
FENGHUA ZHU, Ph.D.[2]
E-mail: fenghua.zhu@ia.ac.cn
[1] Quanzhou Institute of Equipment Manufacturing
Haixi Institutes, Chinese Academy of Sciences
Jinjiang, 362200, China
[2] The State Key Laboratory of Management and Control
for Complex Systems, Institute of Automation
Chinese Academy of Sciences, Beijing 100190, China

Traffic Engineering
Preliminary Communication
Submitted: 30 Jan. 2019
Accepted: 21 Nov. 2019

SOFTWARE-DEFINED ARCHITECTURE FOR URBAN REGIONAL TRAFFIC SIGNAL CONTROL

ABSTRACT

Regional Traffic Signal Control (RTSC) is believed to be a promising approach to alleviate urban traffic congestion. However, the current ecology of RTSC platforms is too closed to meet the needs of urban development, which has also seriously affected their own development. Therefore, the paper proposes virtualizing the traffic signal control devices to create software-defined RTSC systems, which can provide a better innovation platform for coordinated control of urban transportation. The novel architecture for RTSC is presented in detail, and microscopic traffic simulation experiments are designed and conducted to verify the feasibility.

KEY WORDS

regional traffic signal control; heterogeneous environment; software-defined RTSC systems; virtualization;

1. INTRODUCTION

Nowadays, traffic congestion has become a shared challenging problem in many big cities around the world. As the essential component of urban transportation management, traffic signal control has been applied for a very long time in history [1, 2]. Especially, since the 1970s regional traffic signal control (RTSC) platforms, such as TRANSYT, SCOOT, SCATS and so on [3-5], have been rising as the main method of coordinated traffic signal control applied in the cites. However, RTSC systems still face a lot of realistic challenges.

Firstly, with the improvement of refined transportation management, except for traditional road traffic signal controller, new Traffic Signal Control (TSC) devices (e.g. ramp lamp, reversible lane lamp, and variable lane lamp) emerge in the urban area. These devices also require collaborative control, such as traffic control practices in the United States [6] and Europe [7]. Secondly, there are more application requirements for RTSC, e.g. transportation diversion and evacuation in public emergencies or large-scale activities, which are presented in our previous work [8]. Last but not least, the environment for wide-ranging RTSC is usually heterogeneous, especially in the cities with long history. Over the same period the manufacturer of TSC devices installed may be distinct. It is rather difficult for platforms and devices from different manufacturers to dock with each other. Even with national or industry standards like NTCIP [9], only low-level interoperability can be guaranteed actually. Hence, if not impossible, it is formidable to upgrade RTSC systems or TSC devices independently. The heterogeneous problems are so complicated that researchers on RTSC often ignore them and assume that the actual environment is ideal [10-12]. However, obviously, unless the issue is considered during the research process, their innovations are difficult to apply in practice. At present, although there are few related studies, some scholars have begun to pay attention to this issue. For example, Zhang et al. work on traffic signal control of a heterogeneous traffic network with signalized and non-signalized intersections [13].

These realistic challenges place extremely high demands on RTSC systems. However, almost all existing RTSC platforms are closed and have poor compatibility, and they are increasingly unable to meet the needs of urban transportation management. Therefore, the paper proposes a novel software-defined architecture for RTSC, abbreviated as S-RTSC. By adding software virtualization of TSC devices, S-RTSC overcomes the lacks of the existing RTSC platforms, and enables innovation in urban transportation management.

The rest of the paper is organized as follows: Section 2 reviews the architecture of the existing RTSC platforms and their shortcomings. Section 3 proposes the S-RTSC architecture with its composition in detail. In Section 4, traffic simulation is conducted to verify the feasibility. Section 5 summarizes the advantages of S-RTSC.

2. ARCHITECTURE OF THE EXISTING RTSC PLATFORMS

Many countries have developed their own RTSC platforms, including the British TRANSYT system, SCOOT system, Australia's SCATS system, the US RT-TRACS system, Japan's KATENT system, etc. Although each platform adopts different coordinated control strategies, their control structure is similar, as shown in *Figure 1*. The structure can be roughly divided into two parts: traffic signal controller and regional traffic signal control software, two of which are usually connected via Ethernet.

The regional traffic control software, which adopts the centralized control strategies or performs hierarchical control strategies by dividing the city into sub-areas, runs on single or multiple computers. The traffic signal controllers mainly refer to devices that control traffic lights at the intersection. According to the internal communication protocol, the controllers interact with the regional traffic control software directly, accepting upper-level control commands and uploading their running status and traffic flow data collected (if any). The communication protocol for RTSC contains a system of rules that allow two or more entities to transmit information via any kind of variation of a physical quantity. In addition, the protocol should also describe the syntax, semantics, and synchronization of communications, and include sophisticated techniques for detecting and recovering from transmission errors and for encoding and decoding data.

Precisely, the communication protocol becomes the biggest obstacle to development. For commercial reasons, the manufacturers typically do not open the protocols, which has resulted in the inability for others to replace the traffic signal controller or regional traffic control software independently in practice. Over a long time, people have tried to solve this problem by setting communication standards [14]. However, due to a lot of differences in the system design, product logic and control strategies, there is still no universal and easy-to-promote communication standard in the world.

3. S-RTSC ARCHITECTURE

To overcome the aforementioned problems, the paper designs a new architecture S-RTSC, which is implemented with an open and manufacturer-independent method. The crucial feature of S-RTSC is that heterogeneous devices for traffic signal control are virtualized to software-defined objects with the same interfaces. The layered architecture of S-RTSC is presented in *Figure 2*.

The infrastructure layer is composed of various heterogeneous TSC devices. In addition to the main road traffic signal controller, it also includes other types of traffic signal devices, such as controllers for ramp lights, variable lane lights, tidal lane lights and so on. The virtualization layer is by definition

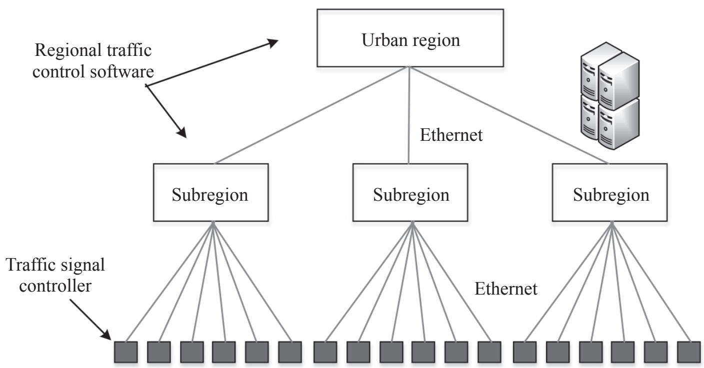


Figure 1 – Control structure of the existing RTSC systems

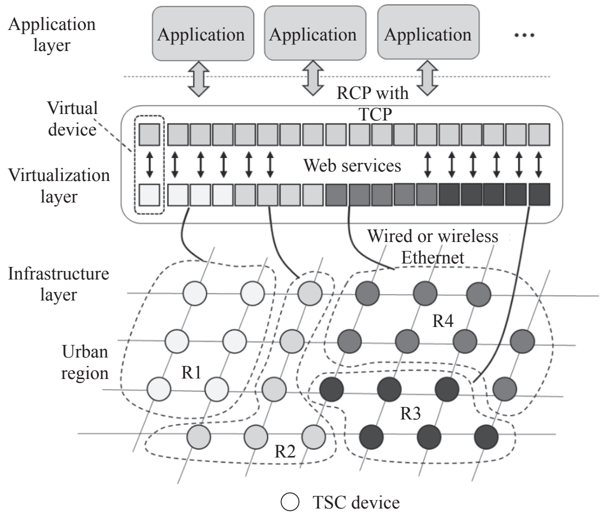


Figure 2 – Layered architecture of S-RTSC

the virtualization of TSC devices, which can be considered as middleware. Each TSC device has a corresponding virtual device.

The way in which TSC devices communicate with their native remote controllers, e.g. wired or wireless Ethernet based on cable or 3G/4G/5G, is reused as the connectivity between the infrastructure layer and virtualization layer. The original functions of remote controllers are transferred to the application layer, which communicates with the virtualization layer by RPC (Remote Procedure Call) with TCP (Transmission Control Protocol). When receiving the request from the application layer, virtual devices are responsible for transmitting it to the corresponding TSC devices through the network. The cross-platform Web Service technology is used as the standard implementation of this part [15]. *Figure 3* demonstrates the software structure of virtual devices. A virtual device consists of a unified interface module and a local web service module. The running of services is based on open standards like XML (eXtensible Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Description, Discovery and Integration). XML is the data format used to contain the data and provide metadata around it. SOAP is used to transfer the data. WSDL is used for describing the services available and UDDI lists what services are available.

The local web service module is responsible for interacting with various TSC devices, so the cooperation of device manufacturers is indispensable. If the manufacturer is willing to provide the (partial) communication protocol, then the native code can be written directly and later packaged into a local service. If the manufacturer is not willing to expose technical details, it can provide a post-encapsulation SDK or Web service (local or remote), which can be further packaged into a local service. The Web service approach enables the manufacturer to implement device virtualization without changing the existing software and hardware products, making it easier for the manufacturers to cooperate. The devices of different type or vendors use distinct communication protocols and methods, so their virtualization involves a little of customization work. Fortunately, this kind of work just needs to be done once and can be copied to other similar devices. Later, regardless of the software upgrade or the

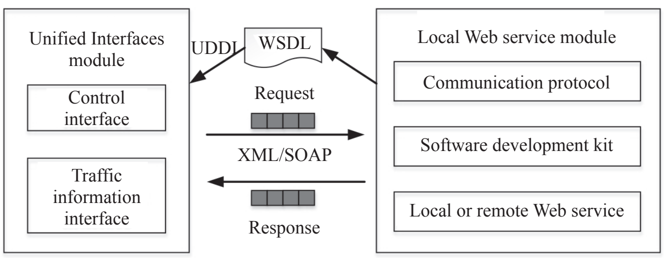


Figure 3 – Software structure of virtual devices

addition or deletion of physical devices, it is only necessary to update or add or delete the corresponding virtual devices.

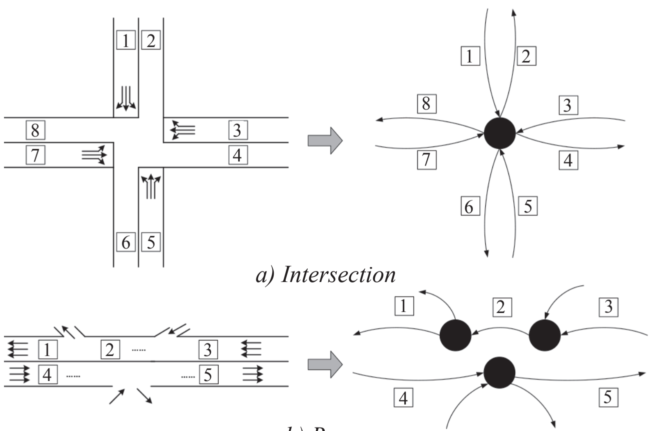
The unified interfaces module is responsible for providing a unified programming interface, including an interface for control and an interface for traffic information. The control interface is called channel table model. A channel table has multiple channel entries, and each channel entry corresponds to a release direction in the traffic signal network. A channel entry item can be represented as a quintuple: <*Time Anchor, Edge Src., Edge Dst., Action, Duration*>. The meaning of each item in the quintuple is listed in *Table 1*. In practice, regardless of the signal light control at the intersection or the indicator light control of the tidal lane, they can be abstracted to release or ban vehicles in a certain travel direction. Thus, the channel table can cover all signal control operations.

The interface for traffic information provides a unified data view of traffic network for RTSC. Urban traffic network is a natural complex network, so the primal approach (treat the intersection as a node and the section as an edge) from the theory of complex network is used to describe the road network [16]. *Figures 4a* and *4b* show two main elements of road network, i.e. intersection and ramp, represented by weighted complex network. The traffic signal control of intersections or ramps in the road network can be represented with attributes of nodes. The reversible lane lights and variable lane lights which affect section capacity can be represented with attributes of edges. Thus, the road transportation state and controllable traffic signal are integrated into one weighted complex network.

Benefit from the consistent programming interface is that the application layer can focus on the algorithm design of traffic signal control. In addition, to coordinate the traffic signal daily, the current RTSC system also needs to play a role in the application scenarios like foreign affairs, large-scale activities, transportation inspection, and public emergencies. Hence, the application layer provides both automatic and manual interfaces to reflect traffic constraints (e.g. stop certain releasing, occupy a certain part of a certain lane) on the network view in a timely manner. Take large-scale activities as an example. At the beginning and in the end of the activity, it is exceedingly easy to cause traffic chaos, accidents and congestion. To avoid these problems,

Table 1 – Items of the channel table

Name	Byte length	Meaning
Time Anchor	4	The trigger time for an action which can be triggered on a periodic basis or in absolute time.
Edge Src.	4	The starting direction of a channel, which is represented by the identifier of road section.
Edge Dst.	4	The target direction of a channel, which is represented by the identifier of road section.
Action	1	Possible actions, including green light, red light, yellow light, green flash, red flash, and yellow flash.
Duration	2	Duration(s) of the action. When the value is 0xFFFF, it means the action is maintained.



a) Intersection

b) Ramp

Figure 4 – An example of road network element represented by weighted complex network (number labels in the figure are identifiers of road section)

specialized applications can be prepared in advance to guarantee smooth arrival and departure paths of the event venue.

4. COMPUTATIONAL EXPERIMENT

To verify the feasibility and advantages of S-RTSC, a prototype has been developed following the architecture of S-RTSC. The system structure is demonstrated in *Figure 5*, where the prototype implements the virtualization and application layers of S-RTSC.

The execution of a typical RTSC algorithm can be usually broken down into three steps: (1) data analysis, (2) decision making, and (3) control plan update. Decision making is the core of the whole algorithm [17] which makes new control plans through a series of calculations based on data analysis. In the prototype, we use the multi-agent method to implement the part of decision making where each TSC device is treated as an agent and all agents together achieve coordinated regional traffic signal control through cooperation and competition [18, 19]. With the development of artificial intelligence, networked and intelligent decision-making algorithms have become a trend. The methods, such as deep learning [20], fuzzy logic [21], rough sets [22], neural networks [23], genetic algorithm [24, 25], particle swarm optimization [26], game theory [27], expert system [28], self-organizing rules [29], dynamic adaptive planning [30], (layered) multi-objective optimization [31, 32], reinforcement learning [33], complex network [34] and their variants have been applied to the design of RTSC algorithm. As benefit from the consistent interface provided by virtualization, these existing research results can be easily ported to S-RTSC applications.

Subject to test conditions, it is not currently possible to test the S-RTSC in practice, so we reconstruct the microscopic traffic simulation engine of artificial transportation system called TransWorld for the experiment [35]. We have started to develop and improve TransWorld using C# independently since 2009. Currently, TransWorld is able to simulate most traffic scenarios in steps of 0.1 to 1 second. Benefit from the used object-oriented programming approach is that all objects involved in traffic (such as vehicles, roads, traffic lights and their controllers) are abstracted and encapsulated into objects, which is very beneficial for the reconstruction. As a computing experimental platform, it has been applied in many major projects such as urban traffic control and large-scale sports events, and plays a crucial role in traffic simulating, analysis and prediction [36]. The simulated traffic scenes and traffic signal controllers are treated as the infrastructure layer of S-RTSC, and the traffic signal controllers in TransWorld are reconstructed to act as the actual ones which can accept remote control by network. Since these controllers are the same and responsible for the traffic lights at the intersections, a simple test protocol based on TCP for adjusting the time plan (see *Appendix* for message content) is sufficient for the experiments, and only one kind of TSC device needs to be converted into virtual devices.

A transportation scene according to the main roads in the second ring of Jinan City, Shandong Province, China, is built, where 419 intersections and 330 road sections with a total length of 570.81 kilometres are selected. The actual road network is

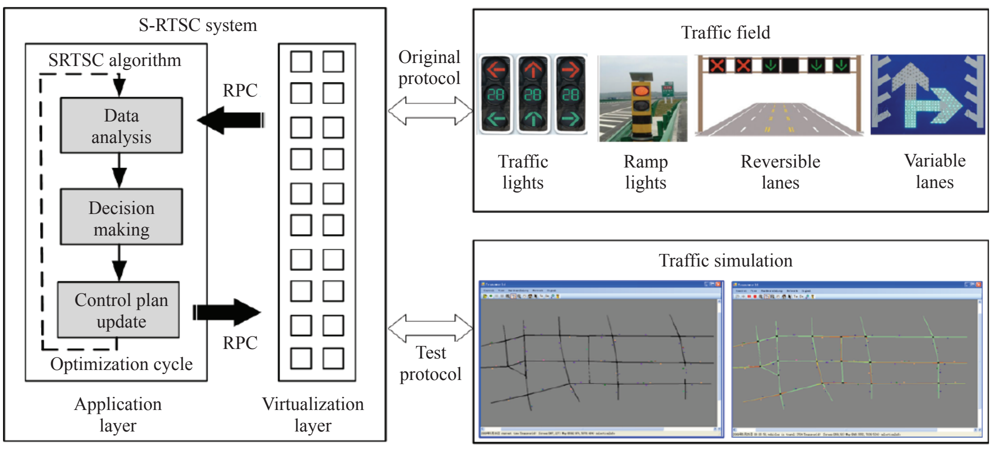


Figure 5 – System structure of the prototype

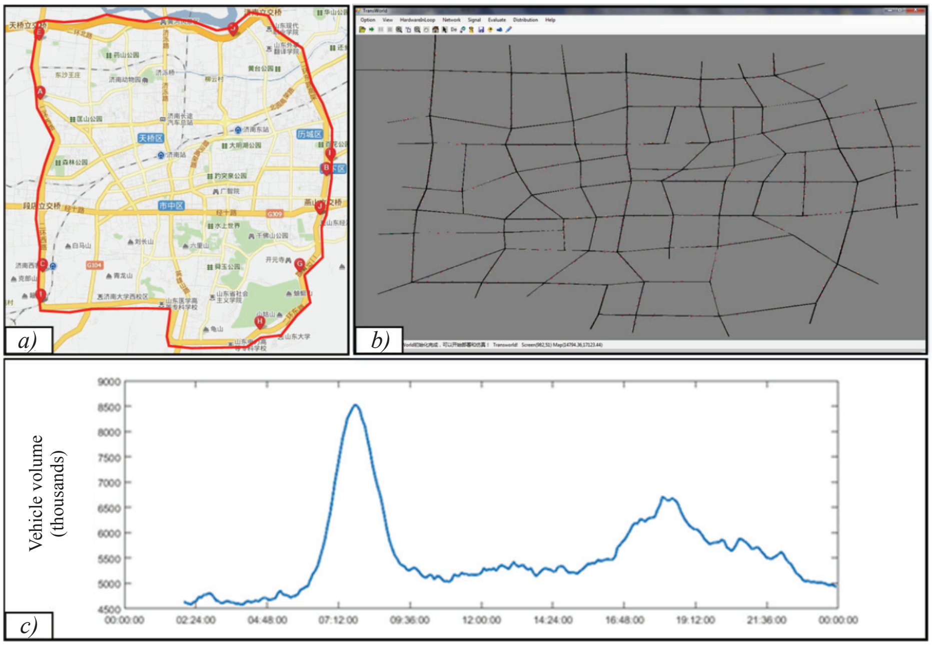


Figure 6 – Computation experiment based on TransWorld

shown in *Figure 6a*, and the corresponding road network constructed in TransWorld is shown in *Figure 6b*. The simulation time is from 00:00-23:59 on a workday. *Figure 6c* shows the change of traffic volume under the prototype. So far, the experiment has verified the feasibility of the new framework. Since the paper does not focus on specific RTSC algorithms, nor are the details of the applied algorithm given here, and neither are the control effects (e.g. average travel speed or delay) discussed or compared further.

In the practical applications, for heterogeneous TSC devices, each offline physical device needs to be converted into online virtual device. In this way, we can build an RTSC system that is loosely coupled to the underlying hardware and supports rapid deployment of applications and easy maintenance. This is the most significant difference between our work and the traditional efforts on communication standards.

5. CONCLUSION

The paper proposes a software-defined RTSC architecture which overcomes the shortages of the traditional ones. By simulation experiments, the feasibility of S-RTSC is verified. In conclusion, the main advantages of S-RTSC are:

1) By adding the virtualization layer, S-RTSC simplifies the deployment and maintenance of RTSC in heterogeneous environment, which is more suitable for large-scale urban traffic control.
2) S-RTSC realizes a unified programming interface, which can be treated as an isolation of software plane and hardware plane, so that new RTSC algorithm can land faster and accelerate application innovation.

The work is an effort to break the closed ecology of RTSC platforms, which can facilitate an orderly competitive market environment. In the future, the virtualization method will be further improved and RTSC applications under the new architecture will be explored.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (No. 61703388, 61603369), the Science and Technology Program of Quanzhou (No. 2018C104R), and the Fujian Science and Technology Plan Guiding Project (No. 2019H0051).

陈松航，博士[1]
E-mail: songhang.chen@fjirsm.ac.cn
张丹，硕士[1]
E-mail: zdan@fjirsm.ac.cn
朱凤华，博士[2]
E-mail: fenghua.zhu@ia.ac.cn
[1] 中国科学院海西研究院泉州装备制造研究所
中国，晋江，362200
[2] 中国科学院自动化研究所复杂系统管理与控制国家重点实验室
中国，北京，100190

软件定义的城市区域交通信号控制架构研究

摘要：

区域交通信号控制（*RTSC*）是一种缓解城市交通拥堵的有效方法。然而，目前*RTSC*平台的生态过于封闭，难以满足城市发展的需要，也严重影响了*RTSC*自身的发展。为此，本文提出了通过虚拟化交通信号控制设备，建立软件定义的*RTSC*系统，从而为城市交通协调控制提供更好的创新平台。论文详细描述了新型*RTSC*系统的体系架构，并设计和开展微观交通仿真实验验证了该架构的可行性。

关键字：

区域交通信号控制；异构；软件定义；虚拟化

REFERENCES

[1] Wang Y, Yang X, Liang H, Liu Y. A Review of the Self-Adaptive Traffic Signal Control System Based on Future Traffic Environment. *Journal of Advanced Transportation.* 2018;(1494): 1-12.

[2] Kotwal AR, Lee SJ, and Kim YJ. Traffic Signal Systems: A Review of Current Technology in the United States. *Science and Technology.* 2013;3(1): 33-41.

[3] Fernandez R, Valenzuela E, Casanello F, et al. Evolution of the TRANSYT Model in a Developing Country. *Transportation Research Part A: Policy and Practice.* 2006;40(5): 386-398.

[4] Selby DL, Powell R, Scoot, et al. Urban Traffic Control System Incorporating SCOOT: Design and Implementation. *ICE Proceedings.* 2015;82(5): 903-920.

[5] Hu P, Tian Z, Dayem A, Yang F. Field Evaluation of SCATS Control System in Las Vegas. *11th International Conference of Chinese Transportation Professionals (ICCTP 2011): Towards Sustainable Transportation Systems, 14-17 August 2011, Nanjing, China*; 2011.

[6] Gordon RL. *Traffic Signal Retiming Practices in the United States*. Transp. Research Board; 2010.

[7] Intelligent Energy Europe, Project Report. *Intelligent Road and Street Lighting in Europe.* The European Association for Creativity & Innovation; 2008.

[8] Zhu F, Chen S, Mao Z, Miao Q. Parallel Public Transportation System and Its Application in Evaluating Evacuation Plans for Large-scale Activities. *IEEE Transactions on Intelligent Transportation Systems.* 2014;15(4): 1728-1733.

[9] NTCIP. Available from: https://www.nema.org/Technical/Pages/NTCIP.aspx

[10] Chu T, Qu S, Wang J. Large-scale Traffic Grid Signal Control with Regional Reinforcement Learning. *IEEE American Control Conference*; 2016.

[11] Shahidehpour M, Li Z, Bahramirad S, Khodaei A. Optimizing Traffic Signal Settings in Smart Cities. *IEEE Transactions on Smart Grid.* 2017;8(5): 2382-2393.

[12] Li P, Mirchandani P, Zhou X. Solving Simultaneous Route Guidance and Traffic Signal Optimization Problem using Space-phase-time Hypernetwork. *Transportation Research Part B: Methodological.* 2015;81: 103-130.

[13] Zhang Y, Su R, Sun C, Zhang Y. Modelling and traffic signal control of a heterogeneous traffic network with signalized and non-signalized intersections. *IEEE Conference on Control Technology and Applications (CCTA)*; 2017. p. 1581-1586.

[14] Patel RK, Rowe E. *An Overview of ITS Standards and Protocols*. Available from: https://www.ite.org/technical-resources/standards/its-standards-and-protocols/

[15] Yue K, Wang X-L, Zhou A-Y. Underlying Techniques for Web Services: A Survey. *Journal of Software*. 2004;15(3): 428-442.

[16] Porta S, Crucitti P, Latora V. The Network Analysis of Urban Streets: A Primal Approach. Environment and Planning B: Urban Analytics and City Science. 2006;33(5): 705-725.

[17] Asthana R, Ahuja NJ, Darbari M, Shukla PK. A Critical Review on the Development of Urban Traffic Models & Control Systems. *International Journal of Scientific & Engineering Research.* 2012;3(1): 1-6.

[18] Chen S, Chen H, Wang DY, Wang S, Zhang J. *A kind of regional traffic optimization control method integrating multiple controllable traffic signals*. China, 201810790876.9 [P]. 2018.07.18.

[19] Balaji PG, Srinivasan D. Multi-Agent System in Urban Traffic Signal Control. *IEEE Computational Intelligence Magazine.* 2010;5(4): 43-51.

[20] Li L, Lv Y, Wang F-Y. Traffic Signal Timing via Deep Reinforcement Learning. *IEEE/CAA Journal of Automatica Sinica*. 2016;3(3): 247-254.

[21] Rahman SM, Ratrout NT. Review of the Fuzzy Logic Based Approach in Traffic Signal Control: Prospects in Saudi Arabia. *Journal of Transportation Systems Engineering and Information Technology.* 2009;9(5): 58-70.

[22] Dong C, Liu Z, and Liu X. Attribute Reduction to Traffic Flow in Area Traffic Control Based on Rough Sets. *Journal of System Simulation*. 2006;18(6): 1524-1528.

[23] Srinivasan D, Choy MC, Cheu RL. Neural Networks for Real-Time Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems.* 2006;7(3): 261-272.

[24] Ceylan H, Bell MGH. Traffic Signal Timing Optimization Based on Genetic Algorithm Approach, Including Drivers' Routing. *Transportation Research Part B: Methodological.* 2004;38(4): 329-342.

[25] Zheng F, van Zuylen HJ, Liu X, Vine SL. Reliability-Based Traffic Signal Control for Urban Arterial Roads. *IEEE Transactions on Intelligent Transportation Systems*. 2017;18(3): 643-655.

[26] Garcia-Nieto J, Olivera AC, Alba E. Optimal Cycle Program of Traffic Lights with Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation.* 2013;17(6): 823-839.

[27] Alvarez I, Poznyak A, Malo A. Urban Traffic Control Problem: A Game Theory Approach. *IEEE Conference on Decision and Control*; 2008. p. 2168-2172.

[28] Wen W. A Dynamic and Automatic Traffic Light Control Expert System for Solving the Road Congestion Problem. *Expert Systems with Applications*. 2008;34(4): 2370-2381.

[29] Wang A, Wu X, Ma B, Zhou Z. Rules Self-Adaptive Control System for Urban Traffic Signal Based on Genetic Study Classification Algorithm. *International*

Conference on Artificial Intelligence and Computational Intelligence; 2009. p. 429-433.

[30] Li T, Zhao D, Yi J. Adaptive Dynamic Programming for Multi-Intersections Traffic Signal Intelligent Control. *IEEE Conference on Intelligent Transportation Systems*; 2008. p. 286-291.

[31] Chen J. A Robust Multi-Objective Compatible Optimization Control Algorithm for Traffic Signal Control. *International Conference on Intelligent Transportation Systems (ITSC)*; 2014. p. 1850-1856.

[32] Zhou Z, Schutter BD, Lin S, Xi Y. Two-Level Hierarchical Model-Based Predictive Control for Large-Scale Urban Traffic Networks. *IEEE Transactions on Control Systems Technology.* 2017;25(2): 496-508.

[33] Abdulhai B, Pringle R, Karakoulas GJ. Reinforcement Learning for True Adaptive Traffic Signal Control. *Journal of Transportation Engineering*. 2003;129(3): 278-285.

[34] Qi W. *The Regional Traffic Signal Control That based on Pinning Control*. Master Thesis. North China University of Technology; 2016.

[35] Wang F-Y. Parallel System Methods for Management and Control of Complex Systems. *Control and Decision*. 2004;9(5): 485-489.

[36] Wang F-Y. Parallel Control and Management for Intelligent Transportation Systems: Concepts, Architectures, and Applications. *IEEE Transactions on Intelligent Transportation Systems*. 2010;11(3): 630-638.

Appendix

Index of bytes		Content	Value
1		Operation type	1
2		Operation object	8
3		Control mode	1
4		Number of time plan	0~255
5		Number of phases	N, 0~127
6		Green time	15~120
7		Green flash time	0~10
8		Yellow time	3~10
9		Red time	3~10
10	Bit 1	Channel 1 (North)	1: released, 0: Not released
	Bit 2	Channel 2 (North)	1: released, 0: Not released
	Bit 3	Channel 3 (North)	1: released, 0: Not released
	Bit 4	Channel 4 (North)	1: released, 0: Not released
	Bit 5	Channel 5 (North)	1: released, 0: Not released
	Bit 6	Channel 6 (North)	1: released, 0: Not released
	Bit 7	Channel 7 (North)	1: released, 0: Not released
	Bit 8	Channel 8 (North)	1: released, 0: Not released
11		Channel 1~8 (East)	ditto
12		Channel 1~8 (South)	ditto
13		Channel 1~8 (West)	ditto
14~5+8N		…	…