Intelligent Cooperative Traffic Light Control Based on Fog Computing and Reinforcement Learning: Postprint

Authors: An Mengmeng, Fan Xiumei, Cai Hanyu

Date: 2018-12-13T00:00:00+00:00

Abstract

The objective of Intelligent Transportation Systems (ITS) is to fundamentally address urban challenges including road safety, vehicle congestion, and environmental pollution. Intersections, as the junctions where roads and vehicles converge, represent the most critical locations for traffic congestion. To mitigate intersection traffic congestion, this study proposes an FRTL (fog reinforcement traffic light) traffic light control model that integrates fog computing and reinforcement learning theory, enabling intelligent cooperative control of traffic lights based on real-time traffic flow information. Fog nodes upload collected real-time traffic flow data to fog servers, which facilitate information sharing within the fog platform. The fog platform formulates traffic light control algorithms by combining processed shared data with Q-learning. The algorithm utilizes detected real-time traffic data to compute appropriate traffic light timing schemes for deployment. Simulation results demonstrate that, compared with traditional time-of-day control methods and arterial traffic light control (ATL), the FRTL control approach enhances intersection throughput, reduces average vehicle waiting time, and achieves the objective of rationally regulating traffic signal timing to alleviate traffic congestion.

Full Text

Preamble

Research on Intelligent Coordinated Control of Traffic Lights Based on Fog Computing and Reinforcement Learning

An Mengmeng, Fan Xiumei†, Cai Hanyu
(Department of Automation & Information Engineering, Xi' an University of Technology, Xi' an 710048, China)

Abstract: The Intelligent Transportation System (ITS) aims to fundamentally address urban challenges including road safety, vehicle congestion, and environmental pollution. Intersections, as the convergence points of roads and vehicles, represent the most severe locations for traffic congestion. To tackle intersection congestion, this paper proposes a FRTL (Fog Reinforcement Traffic Light) control model that combines fog computing with reinforcement learning theory to enable intelligent coordinated control of traffic lights based on real-time traffic flow information. Fog nodes collect real-time traffic flow data and upload it to fog servers, which facilitate information sharing across the fog platform. The fog platform then formulates traffic light control algorithms by integrating processed shared data with Q-learning. These algorithms calculate appropriate traffic signal timing schemes using detected real-time traffic data for final implementation at traffic lights. Simulation results demonstrate that compared with traditional time-phase control methods and arterial traffic light control (ATL), the FRTL approach improves intersection throughput while reducing average vehicle waiting time, achieving the objectives of rationally regulating signal timing and alleviating traffic congestion.

Keywords: intersection; traffic lights; fog computing; reinforcement learning; Q-learning

0 Introduction

The Intelligent Transportation System (ITS) aims to alleviate traffic congestion, reduce accident rates, and mitigate environmental pollution through close coordination among humans, vehicles, and roads. Traffic lights ensure orderly and smooth vehicle passage throughout road networks. However, with annual traffic volume growth, traditional traffic light control methods can no longer dynamically adjust signals based on real-time intersection traffic flow, leading to congestion. Therefore, constructing adaptive traffic control systems to enhance intersection capacity and service quality represents a critical challenge in smart transportation.

To support smoother vehicle movement and achieve better communication efficiency while addressing limitations of traditional vehicular networks in latency, location awareness, and real-time response, this study investigates an integrated fog computing [1] and VANET [2] fog platform. We assume vehicles in the Internet of Vehicles are intelligent, possessing certain computing and communication capabilities. Fog nodes deployed at network edges (e.g., existing roadside infrastructure or intersection traffic lights) can effectively collect, store, process, upload, and exchange traffic data in real time. Collected data is analyzed and computed by fog servers, then combined with Q-learning [4] from reinforcement learning [3] theory to formulate traffic light control algorithms that determine green light durations for each phase, achieving dynamic regulation of signal timing based on real-time traffic flow.

1 Related Research

Current traffic light control methods primarily fall into three categories: fixed-time control [5], actuated control [6], and adaptive control [7]. With increasing vehicle volumes, traditional methods (fixed-time and actuated control) cannot adjust signals according to real-time traffic flow and no longer meet current demands. Adaptive control methods offer flexibility, availability, and optimality that can effectively mitigate congestion, making them a mainstream approach for urban traffic signal regulation.

Implemented adaptive traffic control systems abroad mainly include RHODES (USA), SCOOT (UK), and SCATS (Australia). The RHODES system can obtain necessary traffic state information in advance through predictive models based on stochastic traffic flow characteristics, though without considering delays caused by other traffic flows [8]. SCOOT employs small-step incremental optimization, adjusting timing parameters according to traffic volume to adapt to flow changes in the short term [9]. However, SCOOT requires adjustment of split ratios (the proportion of cycle time available for vehicle movement) and cannot promptly respond to each cycle' s traffic demands. SCATS uses a hierarchical control approach, optimizing signal cycle length, offset, and split ratios based on vehicle saturation and comprehensive volume, but suffers from limited timing schemes and lack of flexibility [10].

The entire traffic network constitutes a large-scale, complex nonlinear system, and reinforcement learning demonstrates excellent learning performance in nonlinear environments, leading to its recent application in traffic light control. Reference [11] introduces a reinforcement learning-based multi-agent system for traffic signal scheduling, studying five intersections with the objective of minimizing queue delay time. Results show that reinforcement learning-based schemes reduce queue delays. Reference [12] frames adaptive traffic control as a Markov game problem, where each intersection' s traffic signals coordinate with adjacent intersections to improve green time utilization, though this approach cannot obtain traffic flow information promptly due to high data transmission latency, preventing real-time signal regulation. Reference [13] proposes an ATL scheme for arterial roads that alleviates arterial congestion but underestimates the impact of non-arterial traffic flow on arterial traffic.

Based on the above analysis, considering that vehicle numbers at intersections vary randomly, real-time traffic flow information acquisition and dynamic signal adjustment are necessary. Fog computing offers low latency and real-time response for collecting traffic information, while reinforcement learning features environmental interaction, autonomous learning, and strong adaptability. Therefore, this paper proposes intelligent coordinated traffic light control based on fog computing and reinforcement learning for multi-intersection scenarios, adjusting traffic signals according to real-time traffic state information to achieve congestion mitigation.

2 Technical Background

This study employs fog computing and reinforcement learning, briefly introduced below.

2.1 Fog Computing

Unlike cloud computing, fog computing is a distributed computing paradigm that extends cloud computing capabilities to the network edge [14], characterized by: (a) low latency and location awareness; (b) wide geographic distribution and strong mobility; (c) large numbers of nodes and sensor networks; (d) real-time information interaction, online analysis, and cloud interaction.

Recent hot research topics based on fog computing include smart cities, intelligent energy management, and industrial wireless sensor networks. These characteristics are fully demonstrated in smart city case studies. For instance, smart agriculture [15]—as part of smart cities—utilizes fog computing through intelligent sensor nodes to monitor plant growth and climate conditions, playing an important role in crop cultivation. Additionally, water resource management [16], greenhouse gas control, and retail automation represent widespread fog computing applications in smart cities. Balancing energy consumption across various domains is crucial. Intelligent energy management [17] for buildings and operational domains (e.g., households or microgrids) is achieved through fog computing platforms by monitoring and metering power consumption of each device, such as household power usage, and managing energy consumption through effective device control like smart lighting and electric vehicle chargers [18].

Large-scale industrial applications typically involve robots, fixtures, machine tools, workpieces, and chemical reactions—i.e., Industry 4.0 [19]. Intelligent factories with industrial wireless sensor networks (IWSN) play vital roles in factory automation, production, and manufacturing. Applying fog computing to IWSN through adaptive operations platforms (AOP) enhances fault model effectiveness [20]. AOP provides asset management, production, and manufacturing functions through fog infrastructure with resources and end-to-end services.

2.2 Reinforcement Learning

Reinforcement learning is a trial-and-error learning method. Unlike supervised learning, it does not instruct the agent on correct actions but evaluates action quality through reward signals fed back from the environment [21]. Figure 1 [Figure 1: see original paper] illustrates the reinforcement learning framework. The core concept involves learning optimal policies from uncertain environmental information, consistently executing actions in a given environment to change its state and obtain reward feedback, which then strengthens the mapping between environmental states and optimal actions [22].

In Q-learning algorithms, rewards obtained after executing an action in a given state are represented as Q-values stored in a table called the Q-table (Table 1). The reinforcement learning system continuously optimizes this table through learning to indirectly obtain optimal policies.

Table 1 Q-table

StateAction | Q(i+1,j) | Q(i+m,j) | Q(i,j+1) | Q(i+1,j+1) | Q(i+m,j+1) | Q(i,j+n) | Q(i+1,j+n) | Q(i+m,j+n) |

The reinforcement learning system updates the Q-table with valuable experience gained from environmental exploration, directing Q-values toward optimality. The update rule is shown in Equation (1):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)]$$

where s_t, a_t, and r_t represent the environmental state, action selection, and reward at time t, respectively. α is the learning rate. When $\alpha = 0$, learning stops; when α approaches 1, learning becomes too rapid, causing premature convergence. γ is the discount factor, $\gamma \in [0, 1]$. As γ approaches 1, the system emphasizes long-term rewards; as γ approaches 0, it focuses on immediate benefits.

3 Traffic Light Management System Based on Fog Computing Platform

The fog platform in this study integrates fog computing and VANET, extending fog computing paradigms to traditional vehicular networks for improved information sharing and communication efficiency while addressing limitations in latency, location awareness, and real-time response.

We design the fog platform architecture in three layers, as shown in Figure 2 [Figure 2: see original paper]:

a) Data Generation Layer. This study assumes all vehicles in the Internet of Vehicles are intelligent devices capable of collecting useful traffic information, making the vehicular network the data generation layer. With real-time computing, transmission, communication, and storage capabilities, intelligent vehicles serve as key data generators. While processing some data for real-time decision-making, vehicles share and upload other data to fog nodes; vehicle detectors also upload detected vehicle counts to fog nodes.

b) Fog Layer. Fog nodes deployed beside intersection traffic signals collect data from intelligent vehicles and vehicle detectors, process it, and upload it to fog servers. Serving as intermediaries between fog servers and intelligent vehicles, fog nodes function not merely as relays or broadcasters but possess computational capabilities to process and store data and make decisions at the fog layer, thereby reducing latency.

c) Cloud Layer. Due to limited fog server storage capacity, when data volume in the fog layer becomes excessive, fog servers upload previously stored data to cloud servers. Cloud servers perform computational analysis on stored data to make optimal decisions from a global perspective, 最典型的例子就是监控、管理、控制道路交通基础设施，实现最佳的城市交通控制。

The specific mechanism for information collection and sharing on the fog platform in vehicular network environments consists of four steps:

a) Vehicles and vehicle detectors send traffic flow information to fog nodes. When a vehicle enters a fog node' s communication range, it communicates with the node every 10 seconds (frequent communication risks communication storms and potential data loss; considering pedestrians cross intersections slower than vehicles, 10-second intervals are appropriate).

b) Fog nodes communicate with each other. Each intersection' s fog node uploads traffic information from intelligent vehicles and detectors to the regional fog server and shares statistics with adjacent intersection fog nodes. Through the traffic light control mechanism (introduced in Section 4), they determine red and green light durations for each intersection to coordinate traffic flow between local and adjacent intersections.

c) Traffic information uploaded from fog nodes to fog servers becomes historical data, which fog servers use to predict traffic flow and improve network mobility.

d) Due to limited storage capacity, fog servers upload temporarily unused data to cloud servers. Cloud servers perform global optimization based on traffic information from various local networks uploaded by fog servers to coordinate traffic flow across the entire network. Figure 3 [Figure 3: see original paper] illustrates the traffic light management control system based on the fog platform (black arrows indicate vehicle travel directions; lightning-shaped arrows represent communication links among cloud, vehicles, fog nodes, and fog servers).

In summary, applying fog computing to traffic signal control offers three main benefits: (a) reduced response time between vehicle applications and traffic lights; (b) pre-processing of collected data at fog nodes before uploading to fog servers, reducing data volume and accelerating communication; (c) storage of some data on fog servers, alleviating cloud server storage burdens.

4 Traffic Light Control Mechanism Based on Q-Learning in Reinforcement Learning

This paper employs Q-learning from reinforcement learning for traffic light control, formulating a control algorithm implemented in the traffic light control module of the FRTL model (introduced in Section 5.1). The algorithm uses traffic flow as states and green light duration as actions, proceeding through these steps:

a) Initialize the Q-table with arbitrary values;

b) Use detected current traffic state as initial state s;
c) Select a green light duration (action) in state s based on Q-value experience;
d) Execute the action, update traffic state, and observe reward r;
e) Update the Q-table according to the update rule;
f) Assign the new traffic state s_{t+1} to state s;
g) Repeat steps c) through f) until Q-values converge.

This study focuses solely on straight-moving traffic flow (east-west and north-south directions). As shown in Figure 4 [Figure 4: see original paper], there are four intersections, each representing an Agent (Agent1, Agent2, Agent3, Agent4). Coordinated control among intersections can be modeled as a Markov Decision Process: $M = (S, A, R)$, where S represents intersection traffic states, A represents traffic control actions (green light duration), and R represents reward feedback after executing control actions.

a) Traffic State. The traffic state comprises vehicles waiting at the intersection plus newly arrived vehicles. Each Agent receives real-time traffic information from fog nodes, including current waiting vehicle counts and newly arrived vehicle counts. Newly arrived vehicle counts are provided by fog servers through calculating the average of the previous three vehicle counts from adjacent intersections. This paper uses `stopVehicle1`, `stopVehicle2`, `stopVehicle3`, `stopVehicle4` to denote waiting vehicles at intersections 1-4 (Agent1-Agent4), and `newVehicle1`, `newVehicle2`, `newVehicle3`, `newVehicle4` for newly arrived vehicles. The traffic state is represented as $S(\text{stopVehicle}, \text{newVehicle})$. For example, for Agent1, $S(10, 6)$ indicates 10 vehicles waiting at intersection 1 and 6 newly arrived vehicles.

b) Traffic Control Actions. There are three control actions: when an intersection is congested and green time has reached the maximum (120s), the action is 1 (maintain green time); when green time cannot allow all waiting vehicles to pass and has not reached 120s, increase green time; when all waiting vehicles can pass with remaining green time, decrease green time. Thus, each agent has three action choices; for agent1, $A_1 = (1, 2, 3)$.

c) Reward. In traffic signal control, agents receive rewards after executing control actions based on traffic conditions. For Agent i ($i = 1, 2, 3, 4$), there are five primary traffic states: $S(\text{stopVehicle} \to -, 0)$ indicates fewer than or equal to 10 waiting vehicles passed with no new arrivals; $S(\text{stopVehicle} \to +, 0)$ indicates more than 11 waiting vehicles passed with no new arrivals; $S(0, \text{newVehicle} \to +)$ indicates no waiting vehicles but fewer than or equal to 5 new arrivals passed; $S(0, \text{newVehicle} \to ++)$ indicates no waiting vehicles but more than 6 new arrivals passed; $S(\text{stopVehicle} \to +, \text{newVehicle} \to ++)$ indicates more than 11 waiting vehicles and more than 6 new arrivals passed.

The reward depends on the transition from current to next traffic state, calculated as:

$$R = \begin{cases} 0 & \text{Other cases} \\ 1 & \text{First case} \\ 2 & \text{Second and third cases} \\ 3 & \text{Fourth and fifth cases} \end{cases}$$

5.1 Model Design

Fog computing offers low latency and location awareness, while reinforcement learning features environmental interaction and strong adaptability. Therefore, this paper proposes the FRTL traffic light control model on a fog computing platform, as shown in Figure 5 [Figure 5: see original paper]:

a) Traffic Information Collection Module. This module collects traffic flow information primarily from intelligent vehicles and vehicle detectors, including waiting vehicle counts, vehicle speeds within fog node range, and current traffic light states, then transmits this information to the fog server module.

b) Fog Node. Fog nodes process traffic information from the collection module. By analyzing and processing collected data, the traffic light control module applies it to Q-learning-based control algorithms to calculate appropriate green times for each phase. Fog nodes also aggregate received traffic information as statistical data uploaded to fog servers.

c) Fog Server. Fog servers perform local traffic control by analyzing and computing statistical traffic information uploaded from fog nodes. During local traffic coordination, fog servers consider both real-time traffic flow and historical information for traffic prediction. The newly arrived vehicle counts in Section 4' s control mechanism are calculated by fog servers based on historical traffic data.

d) Cloud Server. Cloud servers mine traffic information through big data analysis of fog server uploads for global traffic control. Cloud servers can also provide additional services to intelligent vehicles, such as navigation assistance to help control traffic flow.

5.2 Traffic Flow Evaluation Parameters

Traffic lights play a crucial role in traffic flow management. Rationally allocating signal timing ensures vehicles pass intersections as quickly as possible, reducing stranded vehicles. This paper employs three evaluation parameters: intersection throughput, vehicle waiting time, and road saturation.

a) Vehicle Throughput. The number of vehicles passing through an intersection, used to measure road smoothness and traffic flow. Higher throughput indicates smoother intersections.

b) Vehicle Waiting Time. The time vehicles wait at intersections. Higher waiting times indicate slower passage and more severe congestion [23].

c) Vehicle Saturation. Calculated as Saturation $= \frac{\text{Maximum Traffic Volume}}{\text{Maximum Capacity}}$, this important performance indicator measures road service levels, which are divided into four grades: $0 \leq$ Saturation < 0.6 indicates smooth flow and good service; $0.6 \leq$ Saturation < 0.8 indicates slight congestion and fair service; $0.8 \leq$ Saturation < 1 indicates congestion and poor service; Saturation ≥ 1.0 indicates severe congestion and extremely poor service.

6 Experimental Simulation

PTV-VISSIM is a microscopic, time-interval and driving behavior-based traffic simulation modeling tool that analyzes urban network performance under various traffic conditions (e.g., signals, bus stops) and effectively evaluates traffic schemes [24]. VISSIM can generate both online traffic conditions and offline statistics such as travel time and queue length [25]. Therefore, this study employs VISSIM for experimental simulation.

We use VISSIM's offline output statistics (e.g., vehicle queue length, intersection throughput) as MATLAB inputs for experimental simulation. In VISSIM, we configure right-hand traffic rules, 3600s simulation time, 10 time-step simulation precision, 2000 veh/h saturated flow per approach, 90s maximum green time, and 500m intersection spacing.

The reinforcement learning system uses learning rate $\alpha = 0.6$ and discount factor $\gamma = 0.5$. To validate the FRTL model' s effectiveness, we compare it with time-phase control and ATL control [13].

Traffic flow is calculated every 5 time steps (each step = 10s). After running 360 time steps (3600s), we obtain coordinated intersection traffic flow throughput (using intersection 1 as example). Figure 6 [Figure 6: see original paper] shows that the FRTL method allows more vehicles per second through the intersection compared to time-phase and ATL control.

Vehicle waiting time at each time step is recorded. After 360 time steps, total waiting time per intersection is obtained to calculate average total waiting time. Figure 7 [Figure 7: see original paper] illustrates average total waiting time per intersection, showing FRTL reduces average waiting time by 6.4s and 3.1s compared to time-phase and ATL control, respectively.

Figure 8 [Figure 8: see original paper] shows vehicle saturation under three control methods, with two black bold lines indicating saturation threshold values. Time-phase control maintains smooth flow (Level 1 service) when intersection vehicle count ≤ 800; ATL maintains Level 1 service when ≤ 1000 vehicles; FRTL maintains smooth flow even at 1100 vehicles. Comparison shows FRTL achieves lower saturation, maintains road smoothness, and provides better service levels.

We apply one hour of traffic flow data from Xi' an' s Xianning West Road and

Xingqing Road intersection to three control methods. Current green times at this intersection are 45s and 73s in alternating cycles, causing wasted green time even without waiting vehicles. Ignoring vehicle start-up delay, the average time for one vehicle to pass is 2s. Figure 9 [Figure 9: see original paper] shows that with 10 waiting vehicles, FRTL allocates 20s green time while ATL allocates only 15s, causing stranded vehicles due to lower real-time responsiveness. Maximum green time cannot exceed 120s; FRTL reaches maximum green at 60 waiting vehicles, while ATL requires 70 vehicles.

Figure 10 [Figure 10: see original paper] shows stranded vehicle counts. With FRTL, stranded vehicles appear only when waiting vehicles exceed 60; ATL leaves several stranded vehicles each time due to low real-time performance; time-phase control, with fixed green times, leaves stranded vehicles when waiting vehicles reach 30.

7 Conclusion

Fog computing features low latency and real-time information interaction, while reinforcement learning systems can regulate traffic lights based on real-time traffic information collected by fog platforms through self-learned experience, demonstrating strong adaptability. This paper combines fog computing and reinforcement learning to propose the FRTL traffic light control model, aiming to reduce vehicle waiting time, avoid green time waste, and alleviate traffic congestion. In this model, fog nodes upload collected real-time traffic flow information to fog servers, multiple fog servers share information on the fog platform, and the traffic light control module calculates signal timing by combining processed shared data with Q-learning algorithms.

Unlike time-phase and ATL control, the proposed FRTL model considers adjacent intersection conditions. Vehicles, traffic signals, and road infrastructure share real-time traffic information through the fog platform, enabling the control mechanism to formulate appropriate signal timing schemes based on real-time traffic data. Simulation results demonstrate that FRTL effectively alleviates congestion and improves road service levels.

This study assumes smooth traffic flow without accidents. Current research focuses on local road networks with only straight-moving vehicles, though turning vehicles and accident possibilities exist in reality. Future research will consider turning vehicles and fault models for accident handling, achieving global network optimization through local network optimality to alleviate entire network congestion and improve road utilization.

References

[1] Fang Wei. Paradigm shift from cloud computing to fog computing [J]. Journal of Nanjing University of Information Science & Technology, 2016, 8(5): 404-414.

[2] Oulhaci T, Omar M, Harzine F, et al. Secure and distributed certification system architecture for safety message authentication in VANET [J]. Telecommunication Systems, 2017, 64(4): 679-694.

[3] Prabuchandran K J, Hemanth K A N, Bhatnagar S. Multi-agent reinforcement learning traffic signal control [C]//Proc of International IEEE Conference on Intelligent Transportation Systems. Piscataway, NJ: IEEE Press, 2014: 2529-2534.

[4] Khamis M A, Gomaa W. Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework [J]. Engineering Applications of Artificial Intelligence, 2014, 29(3): 134-151.

[5] Tu Xianku. The research of intelligent timing control system for urban traffic signal light [C]//Proc of International Conference on Consumer Electronics, Communications and Networks. Piscataway, NJ: IEEE Press, 2011: 5425-5428.

[6] Lin Xiaohui. Intersection signal control method and model based on full induction control [J]. Modern Transportation Technology, 2015, 12(1): 44-46.

[7] Li Nan, Zhao Guangzhou. Adaptive signal control for urban traffic network gridlock [C]//Proc of International Conference on Control. Piscataway, NJ: IEEE Press, 2016: 1-6.

[8] Zhao Yi, Tian Zong. An Overview of the Usage of adaptive signal control system in the United States of America [J]. Applied Mechanics & Materials, 2012, 178-181: 2591-2598.

[9] Wang Haizhong, Yu Quan, Gu Jiuchun, et al. Research on urban traffic signal control system (2) [J]. Transport Technology Journal, 2004(6): 94-97.

[10] Pascale A, Lam H T, Nair R. Characterization of network traffic processes under adaptive traffic control systems [J]. Transportation Research Part C, 2015, 59(3): 340-357.

[11] Arel I, Liu Chengxi, Urbanik T, et al. Reinforcement learning-based multi-agent system for network traffic signal control[J]. Intelligent Transport Systems Iet, 2010, 4(2): 128-135.

[12] El-Tantawy S, Abdulhai B, Abdelgawad H. Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown toronto [J]. IEEE Trans on Intelligent Transportation Systems, 2013, 14(3): 1140-1150.

[13] Younes M B, Boukerche A. Intelligent traffic light controlling algorithms using vehicular networks [J]. IEEE Trans on Vehicular Technology, 2016, 65(8): 5887-5899.

[14] Alrawais A, Alhothaily A, Hu Chunqiang, et al. Fog computing for the Internet of things: security and privacy issues [J]. IEEE Internet Computing, 2017, 21 (2): 34-42.

[15] Perera C, Jayaraman P P, Zaslavsky A, et al. Sensor discovery and configuration framework for the Internet of Things paradigm [C]//Internet of Things. Piscataway, NJ: IEEE Press, 2013: 94-99.

[16] Perera C, Zaslavsky A, Georgakopoulos D, et al. Sensing as a service model for smart cities supported by Internet of Things [J]. European Trans on Telecommunications, 2014, 25(1): 81-93.

[17] Vatanparvar K, Faruque M A A. Energy management as a service over fog computing platform [C]//Proc of the 6th ACM/IEEE International Conference on Cyber-Physical Systems. New York: ACM Press, 2015: 1-6.

[18] Faruque M A A, Dalloro L, Zhou Siyuan, et al. Managing residential-level EV charging using network-as-automation platform (NAP) technology [C]//Proc of Electric Vehicle Conference. Piscataway, NJ: IEEE Press, 2012: 1-6.

[19] Sasajima H, Ishikuma T, Hayashi H. Future IIOT in process automation —Latest trends of standardization in industrial automation, IEC//TC65 [C]//Society of Instrument and Control Engineers of Japan. 2015: 1-6.

[20] Gazis V, Leonardi A, Mathioudakis K, et al. Components of fog computing in an industrial internet of things context [C]//Proc of IEEE International Conference on Sensing, Communication, and Networking. Piscataway, NJ: IEEE Press, 2015: 1-6.

[21] Gao Yang, Chen Shifu, Lu Xin. A review of reinforcement learning research [J]. ACTA Automatica Sinica, 2004, 30(1): 86-100.

[22] Sutton R S, Barto A G. Reinforcement learning: an introduction [J]. IEEE Trans on Neural Networks, 1998, 9(5): 195-220.

[23] Yit Kwong Chin, Lai Kuan Lee, Bolong N, et al. Exploring Q-Learning Optimization in Traffic Signal Timing Plan Management[C]//Proc of International Conference on Computational Intelligence. Piscataway, NJ: IEEE Press, 2011: 269-274.

[24] Zheng Xiaohong. Application of VISSIM simulation technology in the field of transportation [J]. Management & Technology of SME, 2017(7): 147-148.

[25] Bao Yiting. Modeling and simulation of motion process of right-round motor vehicle and pedestrian conflict at signalized intersection [D]. Nanjing: Nanjing University of Technology, 2017.

Note: Figure translations are in progress. See original paper for figures.