Computer Communications 154 (2020) 324–330

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Traffic signal control for smart cities using reinforcement learning[☆]

Hyunjin Joo[a], Syed Hassan Ahmed[b], Yujin Lim[a,*]

[a] Department of IT Engineering, Sookmyung Women's University, Republic of Korea
[b] Department of Computer Science, Georgia Southern University, USA

ARTICLE INFO

Keywords:
Smart city
Q-learning
Traffic signal control
Traffic congestion

ABSTRACT

Traffic congestion is increasing globally, and this problem needs to be addressed by the traffic management system. Traffic signal control (TSC) is an effective method among various traffic management systems. In a dynamically changing and interconnected traffic environment, the currently model-based TSCs are not adaptive. In addition, with the rise of smart cities and IoT, there is a need for efficient TSCs that can handle large and complex data. To address this issue, this study proposes a TSC system to maximize the number of vehicles crossing an intersection and balances the signals between roads by using Q-learning (QL). The proposed system has a flexible structure that can be modified to suit the changes in the original structure of the intersection.

1. Introduction

Traffic congestion occurs daily in urban areas. It is one of the major challenges that need to be addressed in the transportation system [1]. Traffic congestion is a costly affair as it increases travel time, fuel consumption, and operating costs. It also causes environmental pollution. To address traffic congestion, various studies have been conducted on traffic management systems [2]. Recently, research on intelligent transportation systems (ITS) were conducted, to make traffic management systems safer, efficient and eco-friendly [3,4].

Smart city is the future trend in urban development. It is based on various aspects of Internet of Things (IoT) and Information Communication Technology (ICT). IoT is a technology that connects various things through wireless communication. This applies to almost everything we use in our lives [5]. A smart city comprises of smart homes, smart transportation, green city, smart urban management, and smart tourism, etc. [6] dealing with a lot of data in real-time to communicate with one another. Smart cities need smart and efficient traffic management to overcome the current traffic issues.

Smart traffic management system (TMS) is an important aspect of smart cities to reduce traffic congestion. The traffic signal control (TSC) is considered as the most important and effective means of traffic management. TSC currently uses a fixed-time mechanism, with each signal operating on a fixed time. Earlier, TSC used mathematical models [7,8] and optimization techniques [9–11]. However, for a smart city, the complexity of mathematical models and optimizations is high due to large amounts of data. Moreover, it is difficult to reflect the dynamics of a continuously changing traffic environment. Therefore, we need a TSC system in smart cities that uses AI for traffic management. Among AI techniques, we want to use reinforcement learning to decide the best action in a dynamic environment. Because the traffic environment has stochastic problems, we use Q-learning (QL) which is a model-free reinforcement learning. This study proposes an optimal TSC system that maximizes the number of vehicles crossing an intersection.

The paper is organized as follows. Section 2 introduces related research papers on traffic signal control system related to reinforcement learning. Section 3 presents the system model and the proposed strategy. Section 4 presents and discusses the experimental results. Finally, Section 5 concludes the paper.

2. Related works

Numerous studies have been conducted on intersection traffic management to enhance the intelligence of traffic management systems. However, complicated traffic signal optimization problems cannot be solved using conventional methods. Hence, traffic signals currently use AI techniques such as fuzzy [12], Q-learning (QL) [13–15] and deep Q-learning [16,17]. In fuzzy [12], the optimum extension time of traffic lights is calculated using fuzzy logic. The fuzzy light controller uses membership function to extend the time by identifying traffic flow using two sensors. It is flexible compared to fixed-time controllers. In the fuzzy model, we analyze the dynamic environment and continuously change the model to adapt to the environment. In other words, fuzzy control sentences created for the changing environment are added and applied. These tasks generate a lot of process power. Eventually, it degrades the performance of the system. Hence, researchers experimented with QL to manage traffic flow.

[☆] This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2018R1A2B6002505).

[*] Corresponding author.
E-mail address: yujin91@sookmyung.ac.kr (Y. Lim).

https://doi.org/10.1016/j.comcom.2020.03.005
Received 28 September 2019; Received in revised form 25 February 2020; Accepted 2 March 2020
Available online 3 March 2020

Q-learning (QL) is a reinforcement learning algorithm. It does not use predefined models which makes QL suitable for traffic management problems that change in real-time. In [13], it minimized stop delay to obtain an optimum green light time. Based on the fuzzy rule set, the environment was detected, classified, and the optimum green light time was found using QL. However, the performance was suboptimal for low traffic volumes. In [14], it minimized the queue length of the road to reduce traffic congestion. The traffic light sequence for turning the green light on, was fixed. The green light time was extended or reduced by considering the queue length factor. It was flexible compared to fixed-time traffic signal, but it could not change the traffic light sequence. In [15], it is proposed a QL technique using a cluster to reduce the waiting time for vehicles. Queue length and waiting time were used as parameters. The order of the green light varied depending on the flow of traffic. However, there was a risk of operating only the sections with heavy traffic.

Deep Q-learning (deep QL) is a technique used for high-dimensional inputs dealing with multiple states compared to QL. Usually, it learns Q-function through a deep neural network (DNN). Then, the value-function-based agent selects the optimized control action. In [16], it is proposed to minimize differences in queue length in all directions with parameter. In [17], the difference in total cumulative delays in the current and previous time is minimized. Deep learning models contain multiple layers. However, our signal control problem does not require multiple layers or many states. Hence, this study will explore signal control systems using QL.

This study proposes a new QL algorithm considering throughput and standard deviation of queue lengths as the main parameters. The purpose of this study is to increase the number of vehicles crossing an intersection over a period of time and maintain a balance between roads by adjusting the traffic signals using QL. As mentioned above, the related studies had to redesign Markov model problems (MDP) according to the structure of the intersection. On the other hand, this study defines the action set in the driving directions. Thus, the action set does not change even if the structure of the intersection changes. Therefore, we propose an algorithm that can be easily applied to the various n-way intersections.

3. Proposed algorithm

3.1. Problem definition

Consider the problem of optimizing intersection traffic signal control with the aim of minimizing traffic delay. It is important to handle several vehicles at the intersections simultaneously to minimize traffic delays. Additionally, fair distribution of signals is required. Placing signals only on the sides with heavy traffic may produce good performance, but it would not consider waiting times for drivers on the other side of the road. The parameter which can represent an equitable signal distribution is standard deviation of queue lengths. Queue length refers to the number of vehicles waiting on one side of the road. A small value of standard deviation of queue lengths means that traffic is similar on all sides. This implies that the signal is fairly distributed. Therefore, standard deviation of queue lengths and throughput can be used as parameters to minimize traffic delay and distribute signals reasonably. The problem is defined as follows:

d_{ql}: standard deviation of the queue lengths
t_{inter}: time until the signal is back
l_{signal}: signal length
The model formulation is shown as follows:

$$\max throughput \tag{1}$$

subject to:

$$d_{ql} \leq \varphi \tag{2}$$

$$l_{signal} = \mathrm{c} \tag{3}$$

$$t_{inter} < \varphi'' \tag{4}$$

The model is maximized in Eq. (1). The unit of throughput is defined as the number of vehicles crossing an intersection per hour. In Eq. (2), for equitable signal distribution, the standard deviation of queue lengths should be less than or equal to the specified threshold (φ). Eq. (3) represents that the signal length is constant. In Eq. (4), the time between the end of the green signal in one direction and the start of the next green signal in the same direction is less than the pre-defined threshold (φ'').

3.2. Q-learning

Reinforcement learning is an algorithm that can improve itself through past learning processes [18]. QL is a kind of reinforcement learning that uses a trial-and-error approach to explore the complex and stochastic environment and selects the best behavior based on experience [19]. QL has the concept of state, action, and reward. The situation of the environment is considered as the state, behavior as action and experience as reward. As shown in Eq. (5), an action (a_t) taken in a state (s_t), moves to the next state (s_{t+1}).

$$s_t \overset{a_t}{\rightarrow} s_{t+1} \tag{5}$$

The Q-table is updated using the previous values of ($Q(s_t, a_t)$) for the current state (s_t), action (a_t), reward (r_{t+1}) and the maximum values ($\max_a Q(s_{t+1}, a_{t+1})$) from the new state ($s_{t+1},$) using learning rate ($\eta$) as shown in Eq. (6).

$$Q\left(s_t, a_t\right) \leftarrow Q\left(s_t, a_t\right) + \eta \cdot \left(r_{t+1} + \gamma \cdot \max_a Q\left(s_{t+1}, a_{t+1}\right) - Q\left(s_t, a_t\right)\right) \tag{6}$$

Learning rate and discount factor (γ) affect the convergence and speed of algorithms. The learning rate determines the step size of the movement in the optimal direction at each iteration. The discount factor indicates the importance of the next state. Usually, the range of two parameters is set from 0 to 1. The smaller the learning rate, the more sophisticated it learns but the longer it takes to converge. Conversely, the bigger the learning rate, it converges quickly but can be overshooting. The discount factor is as follows: the closer the discount factor is to 0, the greater the importance of past information. Conversely, the closer it gets to 1, the greater the importance of new information. For example, when the discount factor is 0, nothing is learned and only prior knowledge is utilized for decision. When the discount factor is 1, the agent makes a decision by only the most recent information.

The QL algorithm has two types when determining an action. One is exploitation, the other is exploration. The exploitation chooses an action that maximizes reward obtained from learned information. In other words, it makes the best choice with the learned information. However, it is difficult to find global optimization because exploitation is related to local search. To search more diversity and global optimum, exploration which is related to global search is used. The exploration randomly chooses an action. This is a new attempt to gather various experiences. Rich experience makes it possible to make better choices. The ε-greedy selection is used for exploration. The randomness parameter is ε, which ranges between 0 and 1. In the proposed algorithm, exploration and exploitation have the following effect: The exploitation uses learned information to determine the action which the driving direction with the green light in the next signal. Exploration, on the other hand, assigns a random action that receives a signal.

3.3. Q-learning based traffic signal controller

The purpose of this study is to maximize throughput and minimize the standard deviation of queue lengths. Throughput refers to the number of vehicles processed at the intersections over a period of time, and standard deviation of queue lengths is measured to traffic balance each road direction.

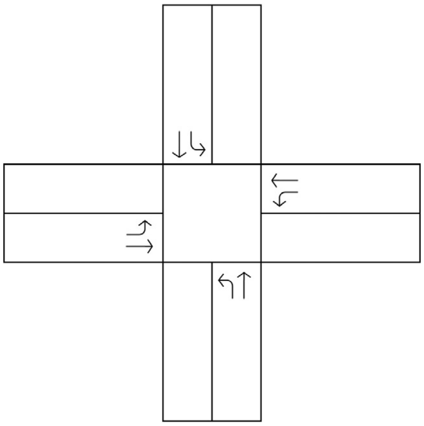


Fig. 1. 4-way intersection.

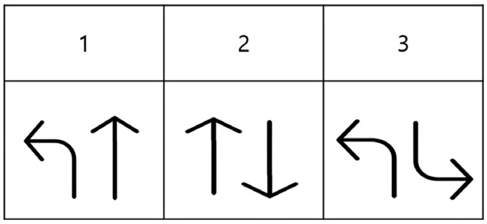


Fig. 2. Action set.

(a) States and actions

States are determined by the number of directions on the road. On one road, there are two directions: left turn and straight. We assume that a right turn is possible in the rightmost straight direction, if necessary. Therefore, an n-way intersection has the 2*n states. The number of directions at one intersection is the number of states. For example, 4-way intersection has 8 states and the 5-way intersection has 10 states. As shown in Fig. 1, a 4-way intersection has a total of 8 directions. The direction with the most number of vehicles becomes the current state. An action that can export the maximum number of vehicle is selected from the action sets that includes the current state direction.

The action sets available for a road are defined in Fig. 2. There are only three action sets available. When an action set is selected, only the direction of the road corresponding to that action will have a green light. The state changes with an n-way intersection, but the action set remains the same.

(b) Rewards

To minimize the delay in an intersection, the reward function is configured with two parameters, i.e., standard deviation (d_{ql}) of the queue lengths of the directions and throughput (tp). A small standard deviation means that all directions at the intersection are similar in length. This ensures a balanced distribution of signals and a balanced length of queues. Intersections with efficient signals handle a large number of vehicles. Therefore, throughput is an important parameter. τ^{tp} is a form of exponential function, whose range τ is between 0 and 1. Larger the tp value, smaller the τ^{tp} value. In other words, throughput is inversely proportional to the τ^{tp} value. α is the adaptive weighting factor, which depends on the arrival of vehicles per hour. α ranges between 0 and 1 and has the form of a sigmoid function. The more the vehicles arrive, the closer it is to 1.

$$f(t) = \alpha \cdot \left(d_{ql}\right) + (1-\alpha) \cdot (\tau^{tp}) \tag{7}$$

$$r_t = log_{\delta}(f(t)) \tag{8}$$

As shown in Eq. (7), a function is expressed in terms of throughput and standard deviation of queue lengths. The minimum $f(t)$ value makes the reward (r) maximum. δ is the base of log function and ranges between 0 and 1.

Table 1
State transition table.

Current state	Action	Next state
s1	a1	{s3, s4, s5, s6, s7, s8}
	a3	{s2, s3, s4, s6, s7, s8}
s2	a1	{s3, s4, s5, s6, s7, s8}
	a2	{s1, s3, s4, s5, s7, s8}
s3	a1	{s1, s2, s5, s6, s7, s8}
	a3	{s1, s2, s4, s5, s6, s8}
s4	a1	{s1, s2, s5, s6, s7, s8}
	a2	{s1, s2, s3, s5, s6, s7}
s5	a1	{s1, s2, s3, s4, s7, s8}
	a3	{s2, s3, s4, s6, s7, s8}
s6	a1	{s1, s2, s3, s4, s7, s8}
	a2	{s1, s3, s4, s5, s7, s8}
s7	a1	{s1, s2, s3, s4, s5, s6}
	a3	{s1, s2, s4, s5, s6, s8}
s8	a1	{s1, s2, s3, s4, s5, s6}
	a2	{s1, s2, s3, s5, s6, s7}

Table 2
Simulation parameters.

Parameter	Value	Unit
The length of a road	5	km
The average speed of vehicles	10	km/h
The length of a vehicle	4.7	m
The distance between vehicles	1.3	m
The learning rate	0.1	–
The discount factor	0.9	–
ε	0.1	–
δ	0.5	–
Epoch	20	–

The proposed MDP diagram is shown in Fig. 3. It shows an interaction between an environment and an agent. The environment denotes a street intersection. At time t, the perceived information which is the queue lengths (ql) and throughput (tp) of all directions in an intersection is sent from the environment to the agent. Then, the agent calculates the reward (r_t) which is received when moving from the state (s_{t-1}) to the state (s_t) and updates the Q-table. Next, the current state (s_t) is set in the direction with the longest queue length based on the perceived information. The agent determines the action (a_t) with the best reward and sends it to the environment. The action indicates the direction (dc) in which the green light is turned on at the current state. Finally, green light turns on in the direction of the street intersection. We assume that the environment is deterministic. Table 1 shows the state transition table with detailed information.

4. Experimental comparison

4.1. Simulation model

The proposed algorithm was tested at a 4-way intersection as shown in Fig. 1. The road was defined as 5 km in length and the speed of the vehicles were 10 km/h on average. Assuming the vehicles were approximately 4.7 m long and the distance between the vehicles was 1.3 m, a single vehicle would take up to 6 m in the queue. The traffic data used in the experiment were extracted using a SUMO simulator [20]. As the traffic environment is stochastic and dynamic, the learning rate was set at 0.1 and ε at 0.1. Through a preliminary experiment, we set δ at 0.5. The discount factor was set at 0.9 because real-time data is more important than historical traffic data in TSC. The detailed simulation parameters are shown in Table 2.

Simulation experiments were conducted to test the performance of the proposed algorithms along with QL. The performance was assessed on three scales: queue length, standard deviation of queue lengths and

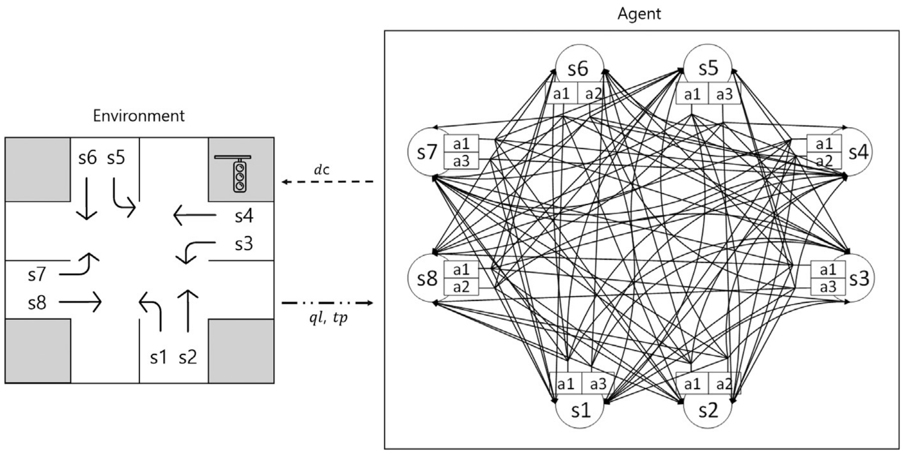
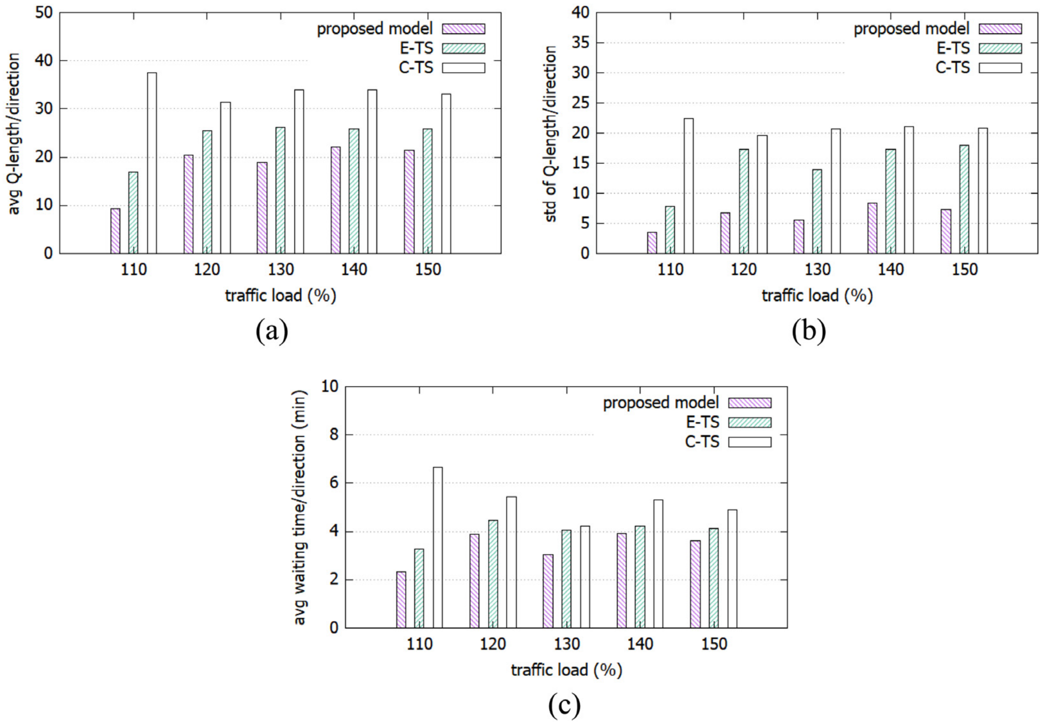


Fig. 3. MDP diagram.



(a)

(b)

(c)

Fig. 4. Performance comparison: (a) queue length, (b) queue length deviation, and (c) waiting time.

waiting time. Queue length indicates the number of vehicles waiting on the road. Standard deviation of queue lengths is measured to balance each road direction; i.e., smaller the value, better the balance. Finally, waiting time is measured as the duration a vehicle stops before crossing the intersection.

4.2. Result and analysis

The proposed model was compared with two other QL models. The first model was based on the order of green light on the road [14]. Hence, it is considered as an upgraded fixed-time traffic light. It decided to extend or shorten the green light time using a QL. The first model is referred as extension traffic signal (*E-TS*) hereafter. The second model controlled traffic signals using a cluster-based QL technique, which will be referred as cluster-traffic signal (*C-TS*) [15]. It handled vehicles in clusters. A cluster crossed the intersection in the green time. A reward was calculated as the sum of queue length and waiting time. For accurate analysis of the results of the experiment, the unit of measurement was adjusted to the unit of direction.

As shown in Fig. 4(a), at 150%, the proposed algorithm had average queue length 25% shorter than *E-TS* and 63% shorter than *C-TS*. As the proposed algorithm has a flexible and undetermined signal system, it demonstrated better performance with increase in the number of arrivals. Fig. 4(b) shows that the proposed algorithm has an average standard deviation value approximately 50% less than *E-TS* and 75% less than *C-TS*. The proposed algorithm has better performance because the proposed algorithm applies standard deviation of queue lengths for calculating rewards. Fig. 4(c), compares the average waiting time per vehicle. On average, the proposed algorithm had approximately 15% less waiting time than *E-TS* and 40% less than *C-TS*. Based on the results of the queue length, standard deviation of the queue length, and average waiting time, it can be confirmed that the proposed algorithm solves the delay of vehicles evenly.

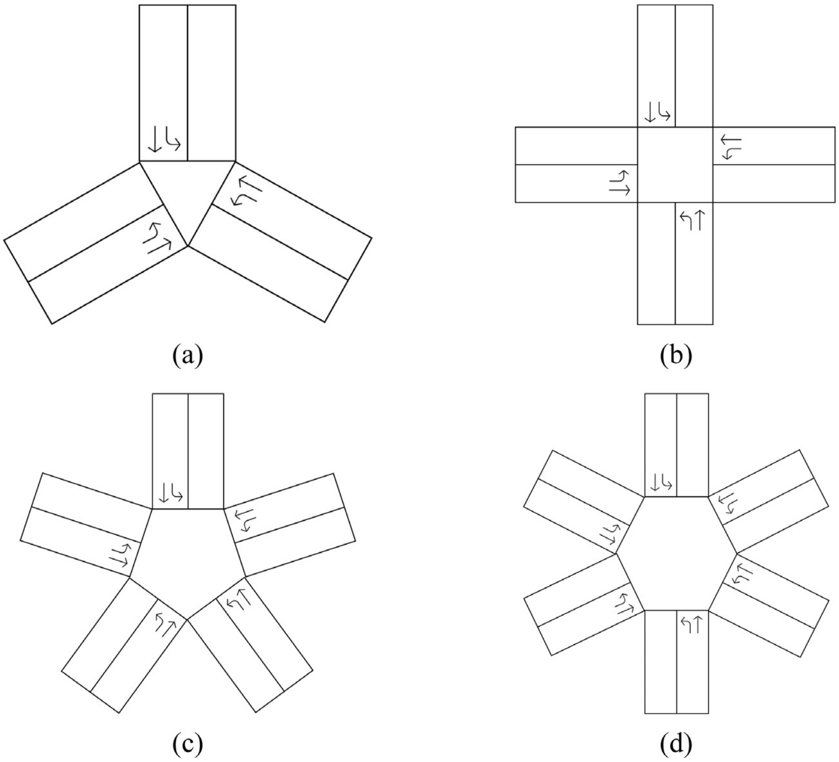


Fig. 5. *n*-way intersection: (a) 3-way, (b) 4-way, (c) 5-way, and (d) 6-way.

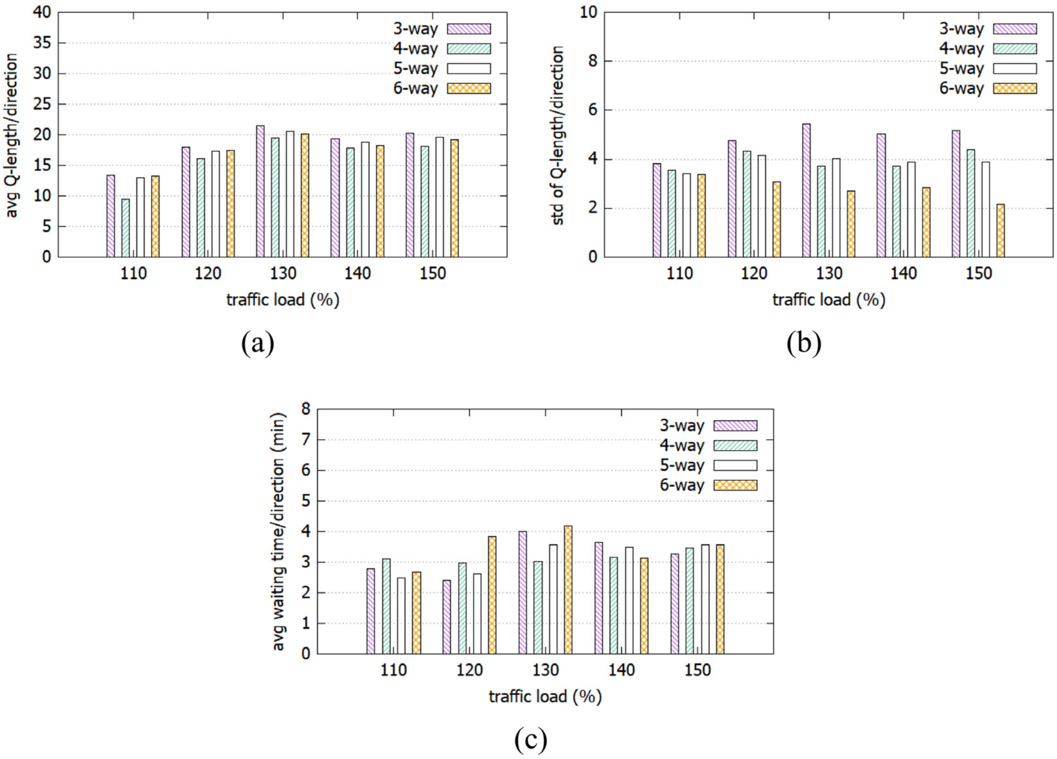
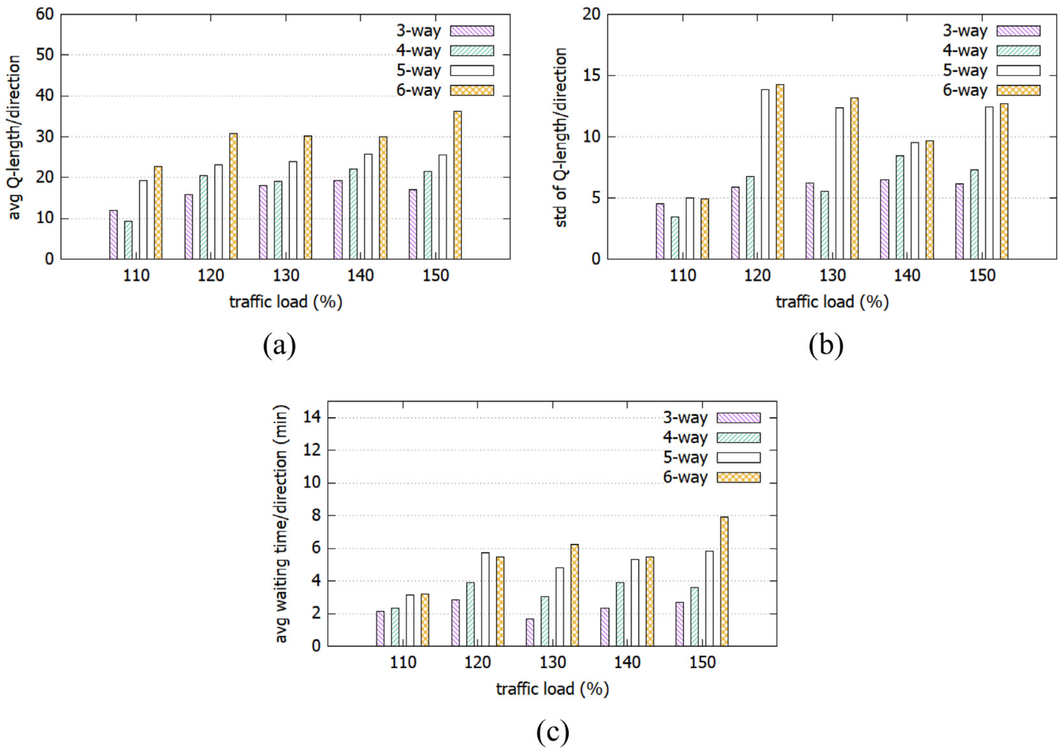


Fig. 6. Hourly road initialization experiment: (a) queue length, (b) standard deviation of queue lengths, and (c) waiting time.

The main advantage of this study is that it has a QL structure that can be applied to various intersections. The performance was analyzed at four different n-way intersections shown in Fig. 5.

The experiment on the *n*-way intersection was divided into two. In the first experiment, the road was initialized every hour, and once in 24 h for the second experiment. The second experiment shows the effects of vehicle delay in the prior time zone on the current time zone. On the other hand, the first experiment is a guide to avoid cumulative delay effect due to hourly initialization. As the values vary depending on the



(a)

(b)

(c)

Fig. 7. 24 h road initialization experiment: (a) queue length, (b) standard deviation of queue lengths, and (c) waiting time.

number of directions, the unit of measurement was adjusted to the unit of direction for accurate analysis.

Fig. 6 shows the results for the hourly road initialization experiment. As shown in Fig. 6(a), the performance of 3, 4, 5 and 6-way intersection are similar. When the traffic load is 110%, all values are distributed between 10 and 15. When the traffic load is 150%, all values are distributed between 18 and 20. As shown in Fig. 6(b), the standard deviation of queue length in the 3-way experiment was greater than other intersections. As 3-way intersection has fewer roads than other intersections, there are some restrictions in the possible action set. For example, the 3-way intersection cannot perform a third action that allows both left turns of different roads in the action set. Inability to perform an action means unable to control traffic in various combinations. Seldom, an unnecessary action combination is selected. Hence, the standard deviation is larger than other intersections. However, the difference in absolute values is not significant. In terms of average waiting time, it exhibited similar results per direction at all intersections. This indicates fair distribution of signals.

Fig. 7 shows the results for the road initialization experiment conducted every 24 h. According to Fig. 7(a), the queue length of 6-way intersection has the largest value compared to other intersections. This is because it has the most number of directions that require a signal in a limited time. It is also affected by accumulated delay due to the highest traffic load. The standard deviation in Fig. 7(b) shows greater values for 5- and 6-way intersections compared to 3- and 4-way intersections especially at 120% and 130%. This suggests that 5- and 6-way intersections calculate more directions. Here, the standard deviation, which indicates the balance on the road, is less balanced than 3- and 4-way intersections. Similarly, Fig. 7(c) shows an increase in the waiting time for 5- and 6-way intersections. Therefore, the proposed technique is not only expandable to various intersection structures but also performs well.

5. Conclusion

This study proposed a traffic signal control system based on QL considering standard deviation of queue lengths and throughput as the main parameters. Compared to other studies using QL, the proposed technique exhibited good performance in terms of standard deviation of queue lengths along with shorter queue length and waiting period. It is interpreted that the traffic signal control system understood the traffic flow and distributed the signals accordingly. This paper studied traffic signal control that is expandable to various intersection structures. As future works, we will study about cooperation among intersections adjacent to each other. As several intersections in a city are interconnected, the traffic load of one intersection affects all the neighboring intersections. We can control the traffic signal more efficiently by sharing information between intersections than by using only local information. The next phase requires a system to control the intersection through communication with adjacent intersections [21–24].

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Hyunjin Joo: Writing - original draft, Software. **Syed Hassan Ahmed:** Conceptualization. **Yujin Lim:** Writing - review & editing, Methodology.

References

[1] A. Bull, Traffic congestion: the problem and how to deal with it, www.eclac.cl/publicaciones.

[2] S. Djahel, R. Doolan, G.-M. Muntean, J. Murphy, A communications-oriented perspective on traffic management systems for smart cities: challenges and innovative approaches, IEEE Commun. Surv. Tutor. 17 (1) (2014) 125–151.

[3] A. Sharif, et al., Internet of things-smart traffic management system for smart cities using big data analytics, in: Proceedings of the 2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP, China, December, 2017.

[4] L. Figueiredo, I. Jesus, J.A.T. Machado, J.R. Ferreira, J.L.M. Carvalho, Towards the development of intelligent transportation systems, in: Proceedings of the 2001 IEEE Intelligent Transportation Systems, USA, August, 2001.

[5] M.Z. Talukder, S.S. Towqir, A.R. Remon, H.U. Zaman, An IoT based automated traffic control system with real-time update capability, in: Proceedings of the 2017 8th International Conference on Computing, Communication and Networking, ICCCNT, India, July, 2017.

[6] K. Su, J. Li, H. Fu, Smart city and the applications, in: Proceedings of the 2011 International Conference on Electronics, Communications and Control, ICECC, China, September, 2011.

[7] P. Serafini, W. Ukovich, A mathematical model for the fixed-time traffic control problem, Eur. J. Oper. Res. 42 (2) (1989) 152–165.

[8] G.F. List, M. Cetin, Modeling traffic signal control using petri nets, IEEE Trans. Intell. Transp. Syst. 5 (3) (2004) 177–187.

[9] D. Zhao, Y. Dai, Z. Zhang, Computational intelligence in urban traffic signal control: a survey, IEEE Trans. Syst. Man Cybern. C 42 (4) (2012) 485–494.

[10] N.H. Gartner, M. Al-Malik, Combined model for signal control and route choice in urban traffic networks, Transp. Res. Rec. J. Transp. Res. (1996).

[11] L. Singh, S. Tripathi, H. Arora, Time optimization for traffic signal control using genetic algorithm, Int. J. Recent Trends Eng. 2 (2) (2009).

[12] I.N. Askerzada, M. Mahmood, Control the extension time of traffic light in single junction by using fuzzy logic, Int. J. Electr. Comput. Sci. 10 (2) (2010).

[13] L. Yongquan, C. Xiangjun, Study on traffic signal control based on Q-learning, in: Proceedings of the 2009 International Conference on Fuzzy Systems and Knowledge Discovery, China, August, 2009.

[14] Y.K. Chin, N. Bolong, A. Kiring, S.S. Yang, K.T.K. Teo, Q-learning based traffic optimization in management of signal timing plan, Int. J. Simul. Syst. Sci. Technol. (2011).

[15] W. Liu, G. Qin, Y. He, F. Jiang, Distributed cooperative reinforcement learning-based traffic signal control that integrates V2X network' dynamic clustering, IEEE Trans. Veh. Technol. 66 (10) (2017).

[16] L. Li, Y. Lv, F. Wang, Traffic signal timing via deep reinforcement learning, IEEE/CAA J. Autom. Sin. 3 (3) (2016).

[17] S.S. Mousavi, M. Schukat, E. Howley, Traffic light control using deep policy-gradient and value-function-based reinforcement learning, IET Intell. Transp. Syst. 11 (7) (2017) 417–423.

[18] G. Li, R. Gomez, K. Nakamura, B. He, Human-centered reinforcement learning: a survey, IEEE Trans. Hum.-Mach. Syst. 49 (4) (2019) 337–349.

[19] D. Pandey, P. Pandey, Approximate Q-learning: An introduction, in: Proceedings of the 2010 Second International Conference on Machine Learning and Computing, India, February, 2010.

[20] Simulation of urban mobility, http://sumo.sourceforge.net/.

[21] H. Ge, Y. Song, C. Wu, J. Ren, G. Tan, Cooperative deep Q-learning with Q-value transfer for multi-intersection signal control, IEEE Access 7 (2019) 40797–40809.

[22] P. Chen, Z. Zhu, G. Lu, An adaptive control method for arterial signal coordination based on deep reinforcement learning, in: Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference, ITSC, New Zealand, October, 2019.

[23] Y.K. Chin, W.Y. Kow, W.L. Khong, M.K. Tan, K.T.K. Teo, Q-learning traffic signal optimization within multiple intersections traffic network, in: Proceedings of the 2012 6th UKSim/AMSS European Symposium on Computer Modelling and Simulation, Malta, November, 2012.

[24] I. Arel, C. Liu, T. Urbanik, A.G. Kohls, Reinforceent learning-based multi-agent system for network traffic signal control, IET Intell. Transp. Syst. 4 (2) (2010) 128–135.