Regular Paper

An Adaptive Traffic Signal Control Scheme
Based on Back-pressure with Global Information

Arnan Maipradit[1,a]   Tomoya Kawakami[2]   Ying Liu[1]   Juntao Gao[1]   Minuro Ito[1]

Abstract: Nowadays traffic congestion has increasingly been a significant problem, which results in a longer travel time and aggravates air pollution. Available works showed that back-pressure based traffic control algorithms can effectively reduce traffic congestion. However, those works control traffic based on either inaccurate traffic information or local traffic information, which causes inefficient traffic scheduling. In this paper, we propose an adaptive traffic control algorithm based on back-pressure and Q-learning, which can efficiently reduce congestion. Our algorithm controls traffic based on accurate real-time traffic information and global traffic information learned by Q-learning. As verified by simulation, our algorithm significantly decreases average vehicle traveling time from 17% to 38% when compared with a state-of-the-art algorithm under tested scenarios.

Keywords: back-pressure, Q-learning, vehicle routing

1. Introduction

1.1 Background

Currently, traffic congestion and increase of vehicle travel time occur because most traffic light control systems use fixed time cycle scheduling [1]. This is led by the algorithms in the control systems not considering real-time or real-situation traffic information. Congestion can be reduced by smartly controlling traffic signals [2]. With the development of technology of intelligent transportation system (ITS) and Internet of Things (IoT) [3], many researchers have adopted such technology to improve the efficiency of transportation. ITS is a traffic management system which uses an intelligent algorithm to reduce vehicle travel time and improve traffic safety.

As an implementation of the intelligent traffic control systems, SCOOT [4] and SCATS [5], [6] were studied. However, these adaptive traffic signals still cannot guarantee a specific performance. In addition, decentralized algorithms are required to realize intelligent traffic control in a practical large scale urban road network. Currently, decentralized traffic control algorithms have been proposed based on back-pressure [7], [8], [9], [10], [11]. Moreover, some back-pressure based algorithms have also been proposed to coordinate different vehicles [12]. In a road network, however, vehicles need time to travel from one road to another road which depends on the vehicle speed and road length. Directly applying a back-pressure algorithm is not appropriate to control traffic as in Ref. [12].

1.2 Contributions

In this paper, we propose an adaptive traffic control algo-

rithm which uses a back-pressure algorithm by considering vehicle traveling time on a road. Specifically, our algorithm controls traffic signals and vehicle routes based on real-time traffic information such as vehicle speed and vehicle position. As a result, our algorithm significantly reduces traffic congestion. In addition, not only based on local traffic information, i.e., every control agent considers information of vehicles around its own junction, we also propose another adaptive traffic control algorithm which uses global traffic information and coordination between different junctions. The latter algorithm controls traffic based on accurate real-time traffic information and local traffic information to global traffic information, where neighboring junction agents exchange traffic information to learn global traffic information.

The proposed algorithms were previously presented in Refs. [13] and [14], respectively. This paper consists of the explanation of the proposed algorithms and experiment results with additional results. Section 2 introduces a road network system assumed in this paper. Section 3 explains our proposed methods, and experiment results are shown in Section 4. Section 5 presents a brief of related works and position of this paper. Finally, this paper is concluded in Section 6.

2. Road Network System

A road system consists of Roads (R) and Junctions (J), where $\mathbb{R} = \{R_1, R_2, R_3, \ldots, R_{max}\}$ denotes roads, $\mathbb{J} = \{J_1, J_2, J_3, \ldots, J_{max}\}$ denotes junctions. It is assumed that each R_i consists of 3 lanes L_{ij}, an example is given in **Fig. 1**. Vehicles of a traffic flow (f) have the same starting road (o) and destination road (d). We define $\mathbb{F}$ as the set of all flows, $\mathbb{O} = \{o(f), f \in \mathbb{F}\}$ as the set of all starting roads, $\mathbb{D} = \{d(f), f \in \mathbb{F}\}$ as the set of all destinations and $\lambda_f(t)$ as the number of vehicles of flow f that enter road network at time slot t.

We define a traffic movement (R_i, R_j) at a junction to be the

1   Nara Institute of Science and Technology, Ikoma, Nara 630–0192, Japan
2   University of Fukui, Fukui 910–8507, Japan
a)   maipradit.arnan.lu0@is.naist.jp
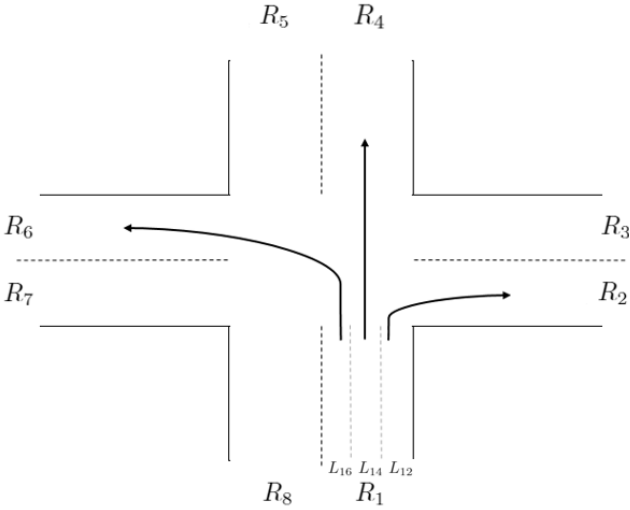
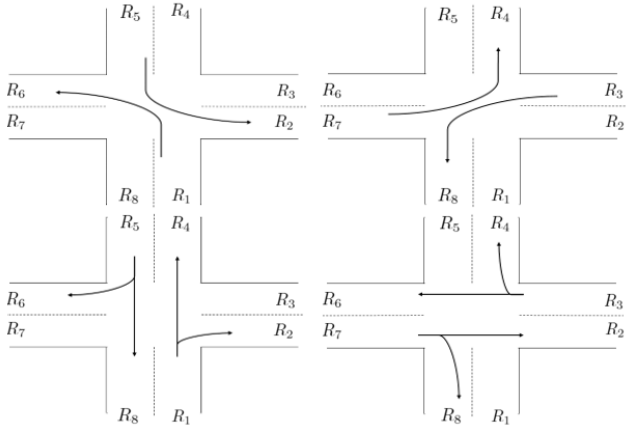**Fig. 1** An example of a junction with roads of three lanes.

Fig. 2 All possible phases at a junction.

process of a vehicle moving from R_i to R_j. We define a traffic phase to include all traffic movements that can happen simultaneously. **Figure 2** shows all possible phases at a junction. For a junction J_a, we define $\mathbb{M}_a$ as the set of all possible movements and $\mathbb{P}_a$ as the set of all possible traffic phases. Traffic signals at junction J_a are controlled by activating a traffic phase p_i^a from $\mathbb{P}_a$.

3. Adaptive Traffic Signal Control Based on Back-pressure with Global Information

In this section, we describe our proposed algorithms based on back-pressure with global information.

3.1 Overview

We have proposed adaptive traffic signal control methods based on back-pressure with global traffic information [13], [14]. Reference [13] describes an adaptive traffic signal control method using real-time traffic information with global traffic information in a road network. We assume all vehicles are self-driving vehicles which are amounted with accurate speed sensors and GPS devices, and can communicate in a timely way with control agents via networks, like vehicle-to-vehicle (V2V), vehicle-to-Road Side Unit (V2R), etc. The control agents are the computer programs placed at each junction to collect information of vehicle speed and vehicle position at every time slot for traffic

control. A vehicle is also able to provide travelling time to control agents. The travelling time includes not only moving time but also waiting time to turn in junctions. At each time slot, every control agent performs the following three tasks sequentially. In addition, the communication loads are greatly increased by control agents if each agent widely exchanges information. Therefore, Ref. [14] employs the approach in which control agents are placed to static equipment on each junction and exchange information only with their neighboring agents. Q-learning is used to estimate global congestion information from limited information. The procedures in our proposed method using the estimated information is as follows;

(1) Task 1. Learning Global Congestion Information:

Control agents exchange congestion level information with their neighboring agents to maintain a table of values $R_{ij}^d(t)$ because the table of values are usually different among neighboring agents. Congestion level information of a road is an index of congestion and defined by the number of shadow vehicles at a shadow queue associated with the road. Based on the exchanged congestion information, the agents update their own congestion estimates based on Q-learning. Through exchanging and updating congestion information, all agents finally obtain global congestion information from recursive definition (1) in Section 3.3. From recursive definition (1), we can see that $R_{ij}^d(t)$ involves all other $R_{ij}^d(t)$ values, thus called global congestion information. Although each road is assigned to two different agents on its junctions, this paper assumes that global congestion information for a specific road obtained by the two assigned agents is not so different because used information is periodically synchronized among neighboring agents. Global congestion information is used in the following two tasks.

(2) Task 2. Traffic Phase Selection:

The agent selects a traffic phase based on the back-pressure algorithm.

(3) Task 3. Vehicle Routing:

After a vehicle passes through the junction and enters the next road under the traffic phase selected in Task 2, the agent determines which lane of that road the vehicle should join. Since each lane determines the vehicle turning direction, i.e., going straight, turning left or turning right, the process of determining lanes for a vehicle to join forms the routing process of that vehicle. The following shadow network is constructed to perform these three tasks.

3.2 Shadow Network

An example of a shadow network is given in **Fig. 3**, where a virtual shadow vehicle in a shadow network corresponds to an actual vehicle in road network, a shadow buffer $\bar{B}_i^d(t)$ corresponds to the beginning part of one real road (a vehicle just passing through a junction will enter this part of the road) and a shadow queue $\bar{Q}_i^d(t)$ corresponds to the end part of one real road (a vehicle running close to the next junction will enter this part of the road).

In the shadow network, whenever a real vehicle enters the road network, a shadow vehicle is generated and enters the shadow network. Furthermore, one more shadow vehicle is generated
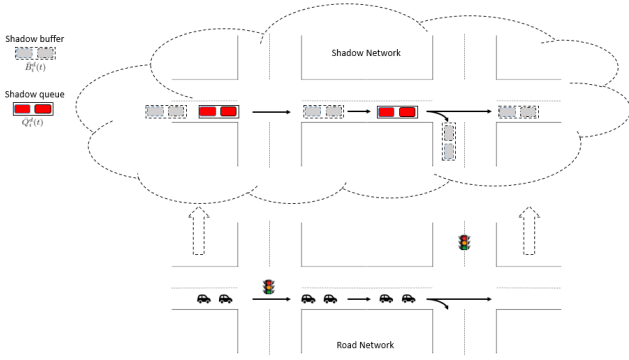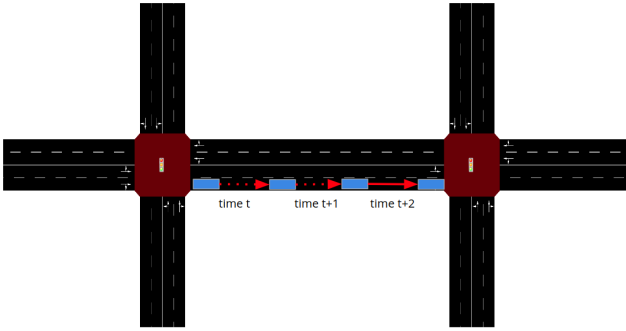
Fig. 3   An example of a shadow network.

Fig. 4   A vehicle needs time to travel across a road.

with probability $\epsilon, 0 < \epsilon < 1$ and also enters the shadow network. This operation makes sure that the algorithm is stable, i.e., queue size will not go to infinite (proper value and detail and Simulation Result and Analysis section of α show in Fig. 12) [12], [15].

When an actual vehicle goes into a road network from starting road R_i at t and wants to go to destination $d \in \mathbb{D}$, a shadow vehicle will also go into $\bar{B}_i^d(t)$. When that vehicle approaches the end part of road R_i, the shadow vehicle first leaves $\bar{B}_i^d(t)$ and then enters $\bar{Q}_i^d(t)$. We say a vehicle approaches the end part of one road if its speed is less than 5 km/h or it is within the range of 100 meters to the next junction.

Similarly, after an actual vehicle destined for destination $d \in \mathbb{D}$ leaves road R_i and goes into the adjacent road R_j at t, a shadow vehicle will leave $\bar{Q}_i^d(t)$ and goes into $\bar{B}_j^d(t)$. **Figure 4** shows vehicles need time to travel from one road to another road which depends on the vehicle speed and road length. Directly applying a back-pressure algorithm without traveling time may decrease the algorithm performance. Movement of virtual shadow vehicles in the shadow network can be seen as control information exchange, based on which an agent performs its three tasks (details are given in the following section).

3.3 Adaptive Traffic Control Algorithm Based on Back-pressure and Q-Learning

Our adaptive traffic control algorithm based on back-pressure and Q-Learning (ARD-BP-Q) is decentralized and the agent at each junction runs the following algorithm independently. At each time slot t, an agent performs the following three steps sequentially.

Step 1. Learning Global Congestion Information

The agent at a junction is responsible for estimating the route congestion level $R_{ij}^d(t)$ for all routes to destination d from road i and by the way of the neighbor road j. Each agent maintains a table $\boldsymbol{R}$ to store the value of $R_{ij}^d(t)$. At the beginning of each time slot, the agent exchanges information of the number of vehicles $\bar{Q}_j^d(t)$ at upstream roads around that junction and the table $\boldsymbol{R}$ with neighboring agents. After exchanging that information, the agent updates its route congestion estimate $R_{ij}^d(t)$ as follows:

$$R_{ij}^d(t) \leftarrow (1 - \alpha)R_{ij}^d(t-1) + \alpha[\bar{Q}_j^d(t) + \gamma \min_k R_{jk}^d(t)] \qquad (1)$$

where α is learning rate and γ is discount factor of Q-learning parameters, $0 < \alpha, \gamma <= 1$. If $R_{ij}^d > C_{max}$, set $R_{ij}^d = C_{max}$ where C_{max} is a positive constant. Bias quantity is a minimum value of estimating the route congestion level that starts from the origin to the destination, each agent then calculates a bias quantity $C_i^d(t)$ as follows:

$$C_i^d(t) = \min_j R_{ij}^d(t) \qquad (2)$$

Finally, the bias quantity $C_i^d(t)$ will be used in Traffic Phase Selection.

Step 2. Traffic Phase Selection

The agents at each junction compute traffic pressure $w_{ij}^d(t)$ for all destinations and traffic movement, traffic pressure is the difference of queue length and bias quantity from the first road to the second road. Traffic pressure in our algorithm ARD-BP-Q (Algorithm 1) is defined as follows:

$$w_{ij}^d(t) = max\{(\bar{Q}_i^d(t) + C_i^d(t)) - (\bar{Q}_j^d(t) + C_j^d(t)), 0\} \qquad (3)$$

Then the agent selects the destination d_{ij}^* that in return maximizes traffic pressure $w_{ij}^d(t)$ defined as follows:

$$d_{ij}^*(t) = \arg\max_d w_{ij}^d(t) \qquad (4)$$

From the above equation, agents define $w_{ij}^{d_{ij}^*(t)}(t)$ as the weight of traffic movement which corresponds to one $d_{ij}^*(t)$ at time slot t.

Finally, the agent selects and activates the phase $p^{a*}(t) \in \mathbb{P}_a$ that releases the most traffic pressure defined as follows:

$$p^{a*}(t) = \arg\max_{p_l^a \in \mathbb{P}_a} \Sigma_{(R_i, R_j) \in p_l^a} w_{ij}^{d_{ij}^*(t)}(t) s_{ij}(t) \qquad (5)$$

where s_{ij} is the number of vehicles that can move from road R_i to road R_j at time slot t.

Step 3. Vehicle Routing

A vehicle will follow the routing probabilities $P_{ij}^d(t)$ based on $\hat{\sigma}_{ij}^d(t)$ defined as follows:

$$P_{ij}^d(t) = \frac{\hat{\sigma}_{ij}^d(t)}{\Sigma_{k: (R_j, R_k) \in \mathbb{M}_a} \hat{\sigma}_{ik}^d(t)} \qquad (6)$$

where $\hat{\sigma}_{ij}^d(t)$ is the estimated value of the expected number of shadow vehicles of destination d that moves from shadow queue $\bar{Q}_i^d(t)$ to shadow buffer $\bar{B}_j^d(t)$ which corresponds to road R_i and R_j. $\hat{\sigma}_{ij}^d(t)$ is updated by the agent of junction J_a for all destination $d \in \mathbb{D}$ and traffic movement $(R_i, R_j) \in \mathbb{M}_a$ as follows :

$$\hat{\sigma}_{ij}^d(t) = (1 - \beta)\hat{\sigma}_{ij}^d(t-1) + \beta\sigma_{ij}^d(t) \qquad (7)$$

where $0 < \beta < 1$. After a vehicle enters road R_i at time slot t, it

will join lane L_{ij} with routing probability $P_{ij}^d(t)$.

Since our goal is to reduce vehicle traveling time, a heuristic is that we should let vehicles with a longer traveling time pass through a junction first. Thus, we also propose the following Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning with Vehicle traveling time (ARD-BP-QV Algorithm 2), which is the same as Algorithm 1 except that traffic pressure is defined as follows:

$$w_{ij}^d(t) = max\{(\bar{V}_i^d(t) + C_i^d(t)) - (\bar{V}_j^d(t) + C_j^d(t)), 0\} \qquad (8)$$

where $\bar{V}_i^d(t)$ is the normalized value of the sum of the traveling time of vehicles in shadow queue $\bar{Q}_i^d(t)$, the normalized value is within range from 50–100. We need to normalize the vehicle traveling time to make it comparable to the quantity of bias $C_i^d(t)$ and $C_j^d(t)$.

4. Evaluation

In this section, we compare the performance of our algorithms with other algorithms below in an open-source simulator SUMO (Simulation of Urban MObility) [16]. **Table 1** shows the compared algorithms in simulation. Our algorithms are ARD-BP-Q and ARD-BP-QV.

4.1 Simulation Setup

We implement a road network that mimics from a simple grid road network and a real Stockholm road network under no roundabout or U-turns situations. The Stockholm road network was given by OpenStreetMap which can export the topology of a road network [17], [18].

4.1.1 Grid Road Network Scenario

The road map we used is given in **Fig. 5**. All roads have different lengths (250–950 meters) and speed limits (60–140 km/h). There are 8 origin and destination pairs $\{(o_1, d_1), (o_2, d_2), (o_3, d_3), \ldots, (o_8, d_8)\}$. All vehicles arrive at the starting roads with the same rates (360–2,520 vehicles/hour) Duration of a slot is configured to be 15 seconds. Shadow vehicle generating probability ϵ is configured to be 0.02 and vehicle routing parameter β is configured to be 0.02.

4.1.2 Stockholm Road Network Scenario

The road network consists of three and four way junctions as shown in **Fig. 6**. All roads have different lengths (400–1,600 meters) and speed limits (60–140 km/h). Roads in this network are bi-directional. There are 6 pairs of origins and destinations $\{(o_1, d_1), (o_2, d_2), (o_3, d_3), \ldots, (o_6, d_6)\}$. All vehicles arrive at the starting roads with the same rates (360–2,520 vehicles/hour). The duration of a slot is configured to be 15 seconds. Shadow vehicle generating probability ϵ is configured to be 0.02 and vehicle routing parameter β is configured to be 0.02.

4.2 Configuration

We define vehicle traveling time to be the time it takes a vehicle to travel from its starting road to its destination. For algorithms AR-BP, ARD-BP, ARD-BP-Q and ARD-BP-QV, parameter $\alpha = 2.5$ (Optimal value of α show in Fig. 14).

During simulations we collect the following data: vehicle speed, number of vehicles in the road network, number of arriving vehicles at destinations and vehicle traveling time. Vehicle traveling time is the time it takes a vehicle to travel from its origin to its destination.

For algorithms FC and SP-BP, we run the simulation for 12,200 seconds. We collect simulation data of vehicles that enter the road network before 7,200 seconds only, because vehicles entering the road network after 7,200 seconds may not arrive at their destinations.
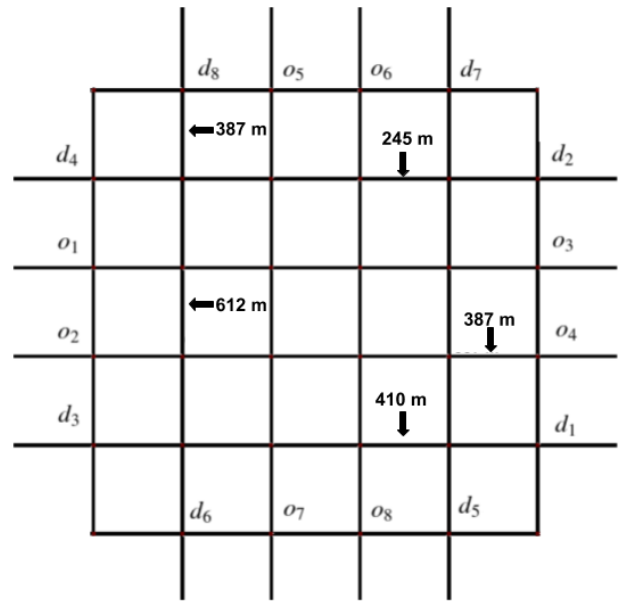

Fig. 5 Grid road network structure that uses in SUMO with 8 pairs of origins and destinations.
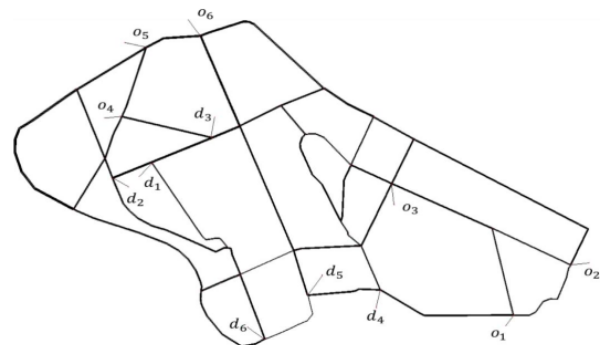

Fig. 6 Road network structure of Stockholm city that uses in SUMO with 6 pairs of origins and destinations.

Table 1 Compared algorithms in simulation.

| Name	Detail
FC	Traffic signal control with fixed-cycles
SP-BP	Back-pressure and shortest path based traffic control algorithm [8]
AR-BP	Back-pressure based adaptive traffic signal control and vehicle routing without real-time control information update [12].
ARD-BP	Back-pressure based adaptive traffic signal control and vehicle routing with real-time control information update
ARD-BP-Q	Adaptive Traffic Control Algorithm Based on Back-Pressure and Global traffic information
ARD-BP-QV	Adaptive Traffic Control Algorithm Based on Back-Pressure and Global traffic information with Vehicle traveling time
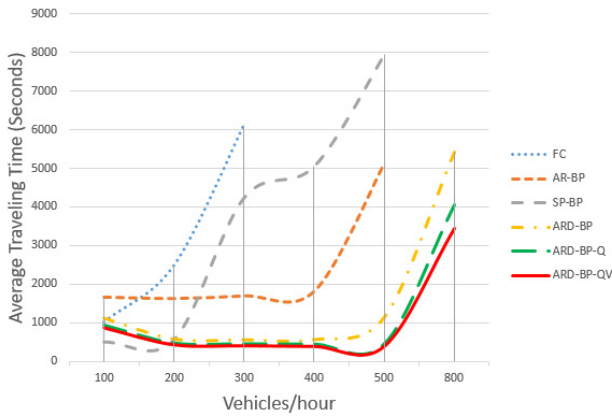
Fig. 7   Average vehicle traveling time under different vehicle arrival rates.
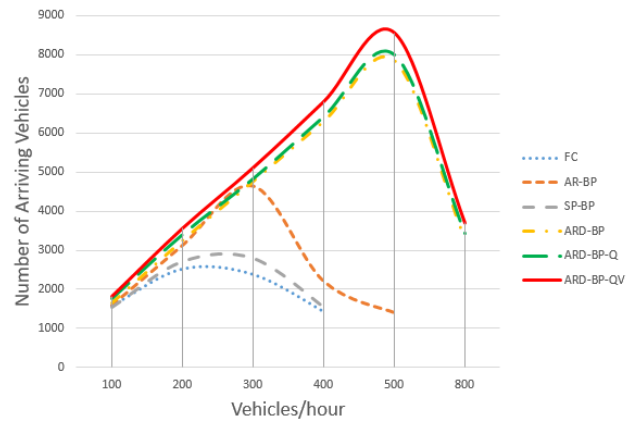

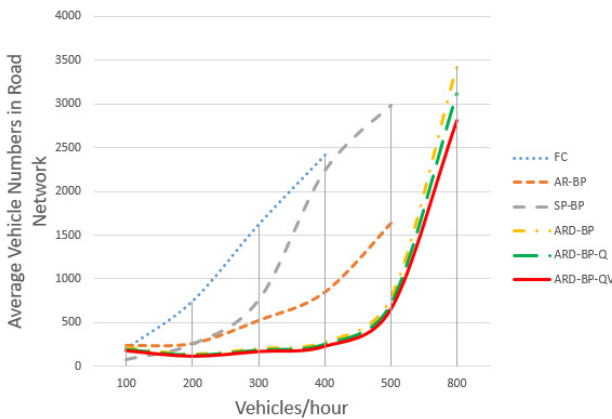Fig. 8   Average number of vehicles in the road network under different vehicle arrival rates.

Fig. 9   Number of vehicles arriving at destinations.

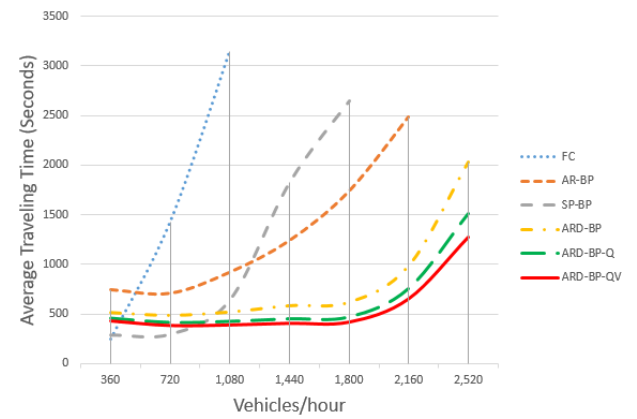Fig. 10   Average vehicle traveling time under different vehicle arrival rates.

Fig. 11   Average number of vehicles in the road network under different vehicle arrival rates.

For algorithms AR-BP, ARD-BP, ARD-BP-Q and ARD-BP-QV, we run the simulation for 18,200 seconds. We collect simulation data of vehicles that enter the road network from 6,000–13,200 seconds only, because these algorithms need time to learn vehicle routing probabilities and reach a stable routing policy.

4.3   Results in Grid Road Network Scenario

In **Fig. 7**, our algorithm ARD-BP-QV achieves almost the lowest average traveling time under different vehicle arrival rates. Compared to ARD-BP, our algorithm ARD-BP-QV decreases the average vehicle traveling time around 36%. Compared to ARD-BP-Q, algorithm ARD-BP-QV decreases average vehicle traveling time around 12%. This indicates that the heuristic of letting vehicles with longer traveling time pass through the junction first is indeed an effective way to reduce vehicle traveling time.

Figure 8 shows simulation results of the average number of vehicles in road network. This figure shows that the number of vehicles in the road network under ARD-BP-QV algorithm is smaller than other algorithms, meaning less traffic congestion.

Figure 9 shows that more vehicles can arrive at destinations under our algorithm ARD-BP-QV, meaning that more vehicles under other algorithms get stuck in the road network.

4.4   Results in Stockholm Road Network Scenario

Also in **Fig. 10**, our algorithm ARD-BP-QV achieves almost the lowest average traveling time under different vehicle arrival rates. Compared to ARD-BP, our algorithm ARD-BP-QV de-
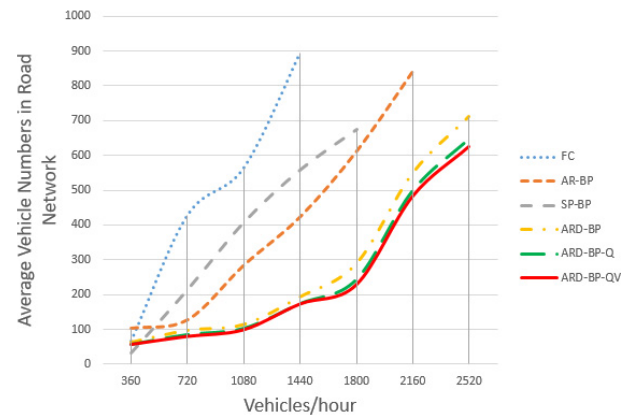
creases the average vehicle traveling time by 17% to 37%. Compared to ARD-BP-Q, algorithm ARD-BP-QV decreases the average vehicle traveling time by 7% to 18%.

Figure 11 shows the simulation results of the average number of vehicles in the road network. This figure shows that the number of vehicles in the road network under the ARD-BP-QV algorithm is smaller than the other algorithms and also in the Stockholm road network scenario.

Figure 12 shows that more vehicles can arrive at destinations under our algorithm ARD-BP-QV. Also in Fig. 12, more vehicles under other algorithms get stuck in the road network.

In the Stockholm road network scenario, we also evaluate the fairness of our algorithm. From **Fig. 13**, we see that most of the
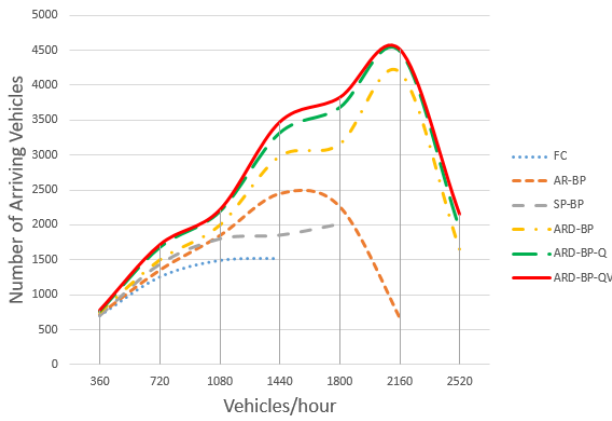
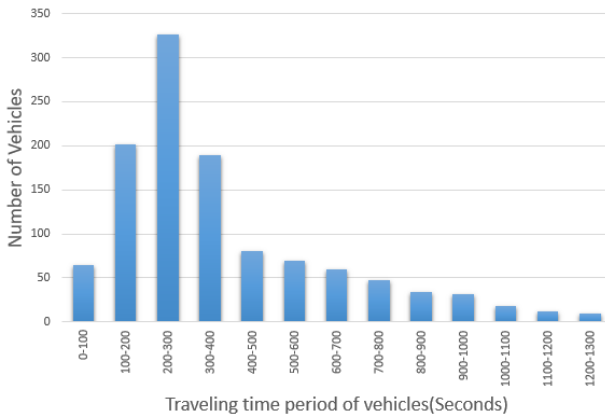**Fig. 12**   Number of vehicles arriving at destinations.

Fig. 13   Histogram of the number of vehicles of different travelling times. Vehicle arrival rate is set to be 1,080 vehicles/hour and the average traveling time is 385 seconds.
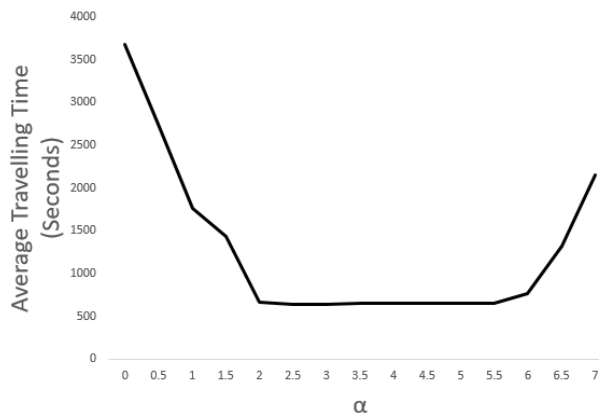

Fig. 14   Performance under parameter α with rate of 450 vehicles/hour.

vehicles arrive at their destinations within 700 seconds, which is less than twice the average traveling time (385 seconds). So, our algorithm is fair for most vehicles.

We also run simulations to check the impact of parameter α to ARD-BP-QV performance. As shown in **Fig. 14**. we need to properly set α in our algorithm to achieve the optimal performance.

Finally, we run a simulation to check our algorithm under the Stockholm road network scenario with both self-driving or human driving vehicles, where all human-driving vehicles follow the shortest path route and the percentage of human-driving vehicles ranges from 10% to 60%. The simulation results are sum-
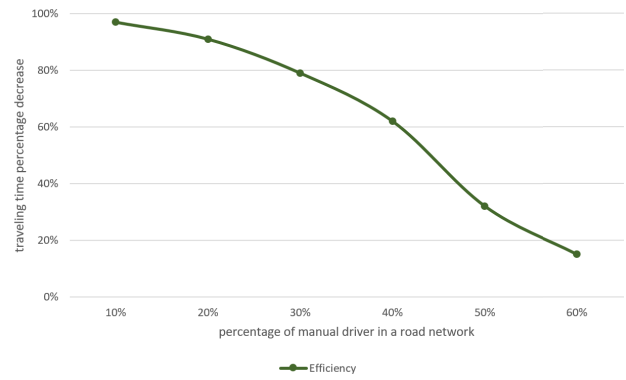

Fig. 15   Vehicle traveling time of ARD-BP-QV under scenarios with both self-driving or human driving vehicles.

marized in **Fig. 15**.

5.   Related Work

Currently, the intelligent traffic control systems have been implemented in an urban road network such as SCOOT [4] and SCATS [5], [6]. These systems use adaptive traffic signals which consider real-time traffic information [19] to become more effective than a fixed cycle signal control. However, these adaptive traffic signals still cannot guarantee global optimality. Genetic Algorithm [20] and Fuzzy Logic Control [21], [22] are also considered as the solution to smartly controlling traffic signals. However, these algorithms are centralized and do not suit a large urban road network which has many entities and requires decentralized algorithms.

Dynamic vehicle routing problems have also been widely studied [23]. The earlier literature only allows vehicles with some minor adjustments of the prior routes [24], [25]. With the development of technology, researchers started using Markov Decision Process to route vehicles dynamically without any prior route [26], [27]. Unfortunately, this method failed being applied in the relatively large scale road networks which exist most in the real-world. To tackle this limitation, an approach based on Approximate Dynamic Programming has been proposed [28], yet all of the solutions above do not integrate with adaptive signal control. Recent researches considered adaptive signal control and dynamic vehicles routing [29], [30]. However, they only focused on providing adaptive route guidance for individual vehicles, not coordinating different vehicles. With the development of self-driving technology, it will be more efficient to coordinate different vehicles to reduce overall traffic congestion.

Recently, decentralized traffic control algorithms have been proposed based on back-pressure [7], [8], [9], [10], [11], as the back-pressure based traffic signal control algorithm shows superior performances to the signal control of fixed time cycles. These back-pressure based traffic control algorithms do not consider the adaptive control of vehicle routes, e.g., a shortest path algorithm easily results in traffic congestion especially during rush hours. Some researches consider jointly controlling traffic signals and vehicle routing [8], [30]. However, these works only focus on giving individual vehicles adaptive route guidance. Coordination between different vehicles will further reduce traffic congestion.

Some back-pressure based algorithms have also been proposed

to coordinate different vehicles [12]. In a road network, however, vehicles need time to travel from one road to another road which depends on the vehicle speed and road length. Directly applying a back-pressure algorithm is not appropriate to control traffic as in Ref. [12].

Compared to the existing back-pressure based algorithms mentioned above, this paper is positioned to an adaptive traffic control algorithm which uses a back-pressure algorithm by considering vehicle traveling time on a road. Specifically, our algorithms control traffic signal and vehicle routes based on real-time traffic information such as the vehicle speed and vehicle position. As a result, our algorithm significantly reduces traffic congestion. In addition, not only based on local traffic information, i.e., every control agent considers information of vehicles around its own junction, this paper covers more efficient traffic control which uses global traffic information and coordination between different junctions.

6. Conclusions

In this paper, we proposed an adaptive traffic control algorithm based on back-pressure with Global traffic information. Our algorithm controls traffic based on accurate real-time traffic information (achieved by using a shadow network) and global traffic information (achieved by using Q-learning). Our algorithm can greatly decrease traffic congestion and is superior to other state-of-the-art algorithms.

Our algorithm is suitable for self-driving vehicles because all vehicles need to completely follow our algorithm. For scenarios with both self-driving vehicles and human-driving vehicles, simulation results show that vehicle traveling time increases as the percentage of human-driving vehicles increase. How to improve algorithm efficiency under these scenarios will be our future work.

References

[1] Li, L., Wen, D. and Yao, D.: A Survey of Traffic Control with Vehicular Communications, *IEEE Trans. Intelligent Transportation Systems*, Vol.15, No.1, pp.425–432 (2014).

[2] Ahmed, E. and Gharavi, H.: Cooperative Vehicular Networking: A Survey, *IEEE Trans. Intelligent Transportation Systems*, Vol.19, No.3, pp.996–1014 (2018).

[3] Hodges, S., Taylor, S., Villar, N., Scott, J., Bial, D. and Fischer, P.T.: Prototyping Connected Devices for the Internet of Things, *IEEE Computer*, Vol.46, No.2, pp.26–34 (2013).

[4] Robertson, D.I. and Bretherton, R.D.: Optimizing Networks of Traffic Signals in Real-Time the SCOOT Method, *IEEE Trans. Vehicular Technology*, Vol.40, No.1, pp.11–15 (1991).

[5] Sims, A.G. and Dobinso, K.W.: The Sydney Coordinated Adaptive Traffic (SCAT) System Philosophy and Benefits, *IEEE Trans. Vehicular Technology*, Vol.29, No.2, pp.130–137 (1980).

[6] Lowrie, P.R.: SCATS, Sydney Co-ordinated Adaptive Traffic System: A Traffic Responsive Method for Controlling Urban Traffic, *Road and Traffic Authority of New South Wales* (1990).

[7] Le, T., Kovács, P., Walton, N., Vu, H.L., Andrew, L.L.H. and Hoogendoorn, S.S.P.: Decentralized Signal Control for Urban Road Networks, *Transportation Research Part C: Emerging Technologies*, Vol.58, pp.431–450 (2015).

[8] Wongpiromsarn, T., Uthaicharoenpong, T., Wang, Y., Frazzoli, E. and Wang, D.: Distributed Traffic Signal Control for Maximum Network Throughput, *Proc. 15th IEEE International Conference on Intelligent Transportation Systems* (*ITSC 2012*), pp.588–595 (2012).

[9] Gregoire, J., Frazzoli, E., de La Fortelle, A. and Wongpiromsarn, T.: Back-Pressure Traffic Signal Control with Unknown Routing Rates, *IFAC Proceedings Volumes*, Vol.47, No.3, pp.11332–11337 (2014).

[10] Gregoire, J., Qian, X., Frazzoli, E., de La Fortelle, A. and Wongpiromsarn, T.: Capacity-Aware Backpressure Traffic Signal Control, *IEEE Trans. Control of Network Systems*, Vol.2, No.2, pp.164–173 (2015).

[11] Kulcsár, B., Ampountolas, K. and Dabiri, A.: Single-Region Robust Perimeter Traffic Flow Control, *Proc. 14th European Control Conference* (*ECC 2015*), pp.2628–2633 (2015).

[12] Zaidi, A.A., Kulcsár, B. and Wymeersch, H.: Back-Pressure Traffic Signal Control With Fixed and Adaptive Routing for Urban Vehicular Networks, *IEEE Trans. Intelligent Transportation Systems*, Vol.17, No.8, pp.2134–2143 (2016).

[13] Liu, Y., Gao, J. and Ito, M.: Back-Pressure Based Adaptive Traffic Signal Control and Vehicle Routing with Real-Time Control Information Update, *Proc. 20th IEEE International Conference on Vehicular Electronics and Safety* (*ICVES 2018*), pp.1–6 (2018).

[14] Maipradit, A., Gao, J., Kawakami, T. and Ito, M.: Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning, *Proc. 22nd IEEE Intelligent Transportation Systems Conference* (*ITSC 2019*), pp.1995–1999 (2019).

[15] Athanasopoulou, E., Bui, L.X., Ji, T., Srikant, R. and Stolyar, A.: Back-Pressure-Based Packet-by-Packet Adaptive Routing in Communication Networks, *IEEE/ACM Trans. Networking*, Vol.21, No.2, pp.244–257 (2012).

[16] Krajzewicz, D., Hertkorn, G., Rössel, C. and Wagner, C.P.: SUMO (Simulation of Urban MObility)—An Open-Source Traffic Simulation, *Proc. 4th Middle East Symposium on Simulation and Modelling*, pp.183–187 (2002).

[17] Haklay, M. and Weber, P.: OpenStreetMap: User-Generated Street Maps, *IEEE Pervasive Computing*, Vol.7, No.4, pp.12–18 (2008).

[18] Haklay, M.: How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets, *Environment and Planning B: Urban Analytics and City Science*, Vol.37, No.4, pp.682–703 (2010).

[19] Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A. and Wang, Y.: Review of Road Traffic Control Strategies, *Proc. IEEE*, Vol.91, No.12, pp.2043–2067 (2003).

[20] Chen, X.-F. and Shi, Z.-K.: Real-Coded Genetic Algorithm for Signal Timing Optimization of a Single Intersection, *Proc. 1st International Conference on Machine Learning and Cybernetics* (*ICMLC 2002*) (2002).

[21] Niittymäki, J.: Installation and Experiences of Field Testing a Fuzzy Signal Controller, *European Journal of Operational Research*, Vol.131, No.2, pp.273–281 (2001).

[22] Pappis, E. and Mamdani, C.P.: A Fuzzy Logic Controller for a Traffic Junction, *IEEE Trans. Systems, Man and Cybernetics*, Vol.7, No.10, pp.707–717 (1977).

[23] Pillac, V., Gendreau, M., Guŕret, C. and Medaglia, A.L.: A Review of Dynamic Vehicle Routing Problems, *European Journal of Operational Research*, Vol.225, No.1, pp.1–11 (2013).

[24] Kenyon, A.S. and Morton, D.P.: Stochastic Vehicle Routing with Random Travel Times, *Transportation Science*, Vol.37, No.1, pp.69–82 (2003).

[25] Secomandi, N. and Margot, F.: Reoptimization Approaches for the Vehicle-Routing Problem with Stochastic Demands, *Operations Research*, Vol.57, No.1, pp.214–230 (2008).

[26] Thomas, B.W. and White, C.C.: Anticipatory Route Selection, *Transportation Science*, Vol.38, No.4 (2004).

[27] Thomas, B.W.: Waiting Strategies for Anticipating Service Requests from Known Customer Locations, *Transportation Science*, Vol.41, No.3, pp.319–331 (2007).

[28] Novoa, C. and Storer, R.: An Approximate Dynamic Programming Approach for the Vehicle Routing Problem with Stochastic Demands, *European Journal of Operational Research*, Vol.196, No.2, pp.509–515 (2009).

[29] Chai, H., Zhang, H.M., Ghosal, D. and Chuah, C.-N.: Dynamic Traffic Routing in a Network with Adaptive Signal Control, *Transportation Research Part C: Emerging Technologies*, Vol.85, pp.64–85 (2017).

[30] Kim, S., Lewis, M.E. and White, C.C.: Optimal Vehicle Routing with Real-Time Traffic Information, *IEEE Trans. Intelligent Transportation Systems*, Vol.6, No.2, pp.178–188 (2005).

Arnan Maipradit   received a B.E. degree from Kasetsart University in 2015 and M.I. from Nara Institue of Science and Technology in 2019 and is also continuing with a Ph.D. at Nara Institue of Science and Technology. His research interests include ITS, IoT, and Mobile computing.

Tomoya Kawakami   received his B.E. degree from Kinki University in 2005 and his M.I. and Ph.D. degrees from Osaka University in 2007 and 2013, respectively. From 2007 to March 2013 and from July 2014 to March 2015, he was a specially appointed researcher at Osaka University. From April 2013 to June 2014, he was a Ph.D. researcher at Kobe University. From April 2015 to February 2020, he was an assistant professor at Nara Institute of Science and Technology. Since March 2020, he has been a senior assistant professor at the University of Fukui. His research interests include distributed computing, rule-based systems, and stream data processing. He is a member of the IEEE.
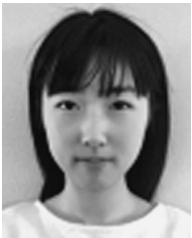
Ying Liu   received her B.S. degree from Sichuan Agricultural University, China and her M.I. degree from Nara Institute of Technology and Science, Japan.

Juntao Gao   received his B.S. and M.S. degrees both in Computer Science from Xidian University, Xi'an, China, in 2008 and 2010, respectively, and received his Ph.D. degree from Graduate School of System Information Science and Technology, Japan. His research interests are in the areas of performance modeling and analysis, stochastic optimization and control in wireless networks, queueing theory and its applications.

Minoru Ito   received his B.E., M.E., and Ph.D. degrees in information and computer sciences from Osaka University in 1977, 1979, and 1983, respectively. He joined the faculty of Osaka University in 1979. Since 1993, he has been a professor of the Graduate School of Information Science at Nara Institute of Science and Technology. He is a member of IEICE, ACM, and IEEE.