# Evaluation of an Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning

Arnan Maipradit[1,a)]   Tomoya Kawakami[1,b)]   Juntao Gao[1,c)]   Minuro Ito[1,d)]

**Abstract:** Traffic congestion causes significant problems such as longer travel time, energy consumption, and air pollution. Currently, we have proposed an adaptive traffic control algorithm based on back-pressure and Q-learning to efficiently reduce congestion. In this paper, we evaluate our proposed method using the road network simulated by a real structure. The simulation results show that our algorithm significantly decreases average vehicle traveling time from 17% to 37% compared with state-of-the-art algorithm.

## 1. Adaptive Traffic Control Based On Back-Pressure And Q-Learning

Our algorithm uses real-time traffic information and global traffic information in road network.

Each junction has a control agent that collects information of vehicle speed and vehicle position every time slot for traffic control. At each time slot, every control agent performs the following three tasks sequentially. Task 1 (Learning Global Congestion Information): It exchanges congestion level information with neighboring agents. Based on exchanged congestion information, the agent updates its own congestion estimate based on Q-learning. Through this kind of congestion information exchange and update, all agents will finally obtain global congestion information which can be used in the following two tasks. Task 2 (Traffic Phase Selection): The agent selects a traffic phase based on back-pressure algorithm. Task 3 (Vehicle Routing): After a vehicle passes through the junction and enters 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 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 three tasks.

## 2. Shadow Network

An example of shadow network is given in Fig. 1, where a virtual shadow vehicle in shadow network corresponds to an actual vehicle in road network, a shadow buffer corresponds to the beginning part of one real road (a vehicle just passing through a junction will enter this part of road) and a shadow queue corresponds to the end part of one real road (a vehicle running close to next junction will enter this part of the road).


1 Division of Information Science Nara Institute of Science and Technology Nara, Japan

a) maipradit.arnan.lu0@is.naist.jp

b) kawakami@is.naist.jp

c) gaojuntao223@gmail.com

d) ito@is.naist.jp


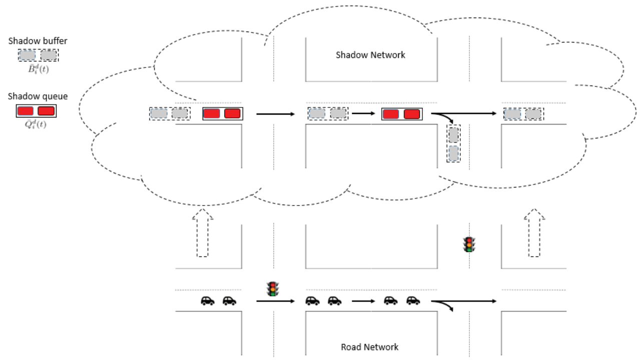


**Fig. 1** An example of a shadow network.

## 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 agent at each junction runs the following algorithm independently.

### 3.1 Task 1 Learning Global Congestion Information

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

$$R^d_{ij}(t) \rightarrow (1-\alpha)R^d_{ij}(t-1) + \alpha[\bar{Q}^d_j(t) + \gamma \min_k R^d_{jk}(t)] \quad (1)$$

where $\alpha$ and $\gamma$ are Q-learning parameters, $0 < \alpha, \gamma <= 1$. If

 

$R_{ij}^d > C_{max}$, set $R_{ij}^d = C_{max}$, $C_{max}$ is a positive constant. Each agent then calculates a bias quantity $C_i^d(t)$ as follows:

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

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

### 3.2 Task 2 Traffic Phase Selection

The agents at each junction compute traffic pressure $w_{ij}^d(t)$ for all destinations and traffic movement. Traffic pressure in our algorithm ARD-BP-Q (Algorithm 1) is defined as follows:

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

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

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

From 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:

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

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

### 3.3 Task 3 Vehicle Routing

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

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

where $\hat{\sigma}_{ij}^d(t)$ is the estimated value of 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 :

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

where $0 < \beta < 1$. After 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 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 with Algorithm 1 except that traffic pressure is defined as follows:

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

where $\bar{V}_i^d(t)$ is the normalized value of the sum of 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 vehicle traveling time to make it comparable to the quantity of bias $C_i^d(t)$ and $C_j^d(t)$.

## 4. Simulation Setup and Result

In this section, we compare the performance of our algorithm with other algorithms in an open-source simulator SUMO (Simulation of Urban MObility) [1]. We implement road network that mimic from a real Stockholm road network which, given by OpenStreetMap that can export topology of road network [2,3].

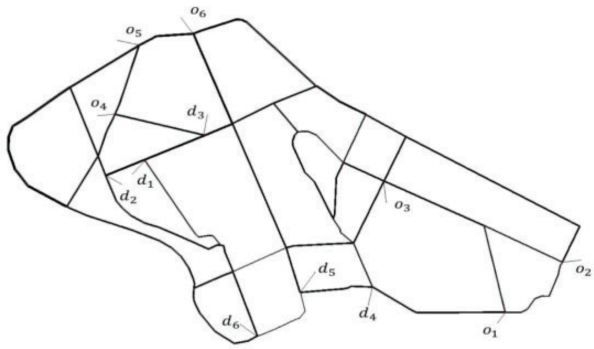


**Fig. 2** Road network structure of Stockholm city that use in SUMO with 6 pairs of origin and destination.

- ◻ Traffic signal control with fixed-cycles (FC)
- ◻ Back-pressure and shortest path based traffic control algorithm (SP-BP) [4]
- ◻ Back-pressure based adaptive traffic signal control and vehicle routing without real-time control information update (AR-BP) [5].
- ◻ Back-pressure based adaptive traffic signal control and vehicle routing with real-time control information update (ARD-BP) [6].
- ◻ Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning (ARD-BP-Q).
- ◻ Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning with Vehicle traveling time (ARD-BP-QV).

In Fig. 3, 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 average vehicle traveling time by 17% to 37%. Compared to ARD-BP-Q, algorithm ARD-BP-QV decreases average vehicle traveling time by 7% to 18%. This indicates that the heuristic of letting vehicles with longer traveling time pass through junction first is indeed an effective way to reduce vehicle traveling time.

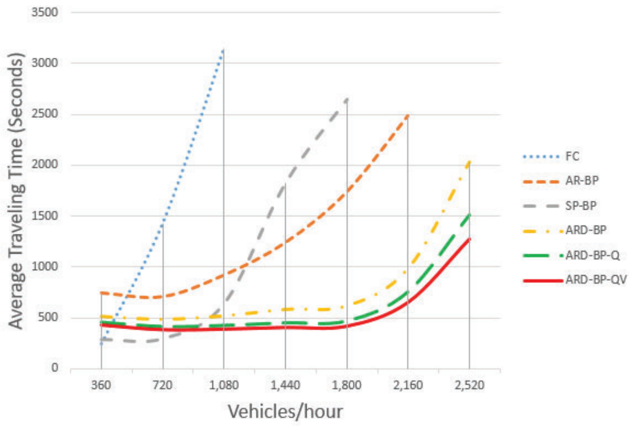


**Fig. 3** Average vehicle traveling time under different vehicle arrival rates.

Fig. 4 shows simulation results of average number of vehicles in road network. This figure shows that the number of vehicles in road network under ARD-BP-QV algorithm is smaller than other algorithms, meaning less traffic congestion.

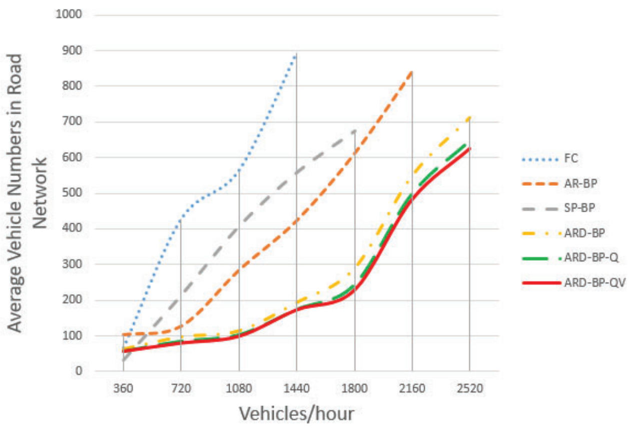


**Fig. 4** Average number of vehicles in road network under different vehicle arrival rates.

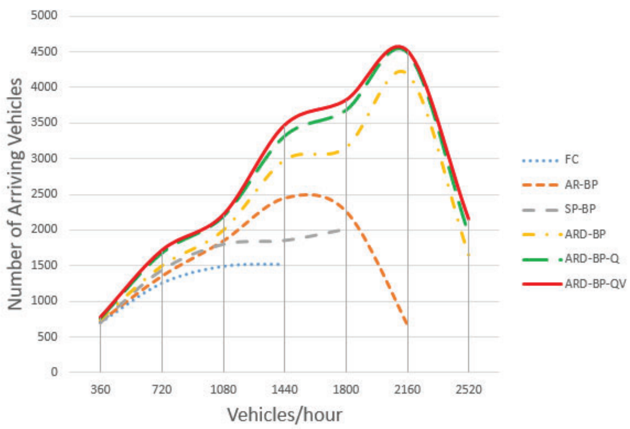


**Fig. 5** Number of vehicles arriving at destinations.

Fig. 5 shows that more vehicles can arrive at destinations under our algorithm ARD-BP-QV, meaning that more vehicles under other algorithms get stuck in road network.

We also evaluate the fairness of our algorithm. From Fig. 6, we see that most of 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.

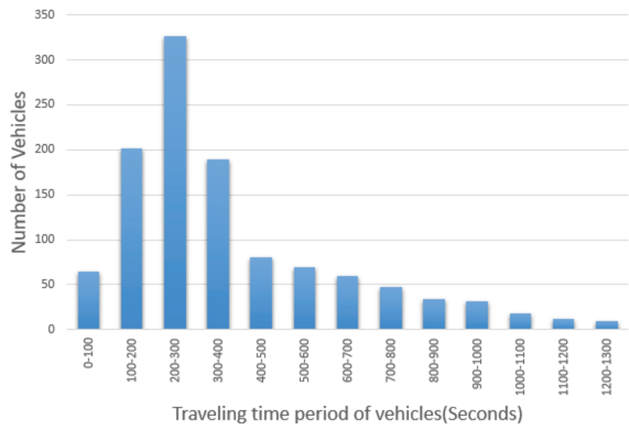


**Fig. 6** Histogram of number of vehicles of different travelling time. Vehicle arrival rate is set to be 1080 vehicles/hour and the average traveling time is 385 seconds.

## 5. Conclusions

In this paper, we proposed an adaptive traffic control algorithm based on back-pressure and Q-learning. Our algorithm controls traffic based on accurate real-time traffic information (achieved by using 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.

## References

[1] D. Krajzewicz, G. Hertkorn, C. Rssel and C, P. Wagner, SUMO (Simulation of Urban MObility)-an open-source traffic simulation, in Proceedings of the 4th middle East Symposium on Simulation and Modelling, 2002, pp. 183–187.

[2] M. Haklay, "How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets," Environment and Planning B: Planning and Design, vol. 37 no. 4, pp. 682–703, 2010.

[3] M. Haklay, P. Weber, "OpenStreetMap: User-Generated Street Maps," IEEE Pervasive Computing vol. 7, no. 4, pp. 12–18, 2008.

[4] T. Wongpiromsarn, T. Uthaicharoenpong, Y. Wang, E. Frazzoli and D. Wang, Distributed traffic signal control for maximum network throughput, in Proc. IEEE ITSC, pp. 588–595, 2012.

[5] A. Zaidi, B. Kulcsr and H. Wymeersch, Back-pressure traffic signal control with fixed and adaptive routing for urban vehicular networks, IEEE Transactions on Intelligent Transportation Systems, vol.17, no.8, pp. 2134–2143, 2016.

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

[7] A. Maipradit, J. Gao, T. Kawakami, and M. Ito, Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning, Proc. of the 22nd IEEE Intelligent Transportation Systems Conference (ITSC 2019), 6 pages, 2019 (to appear).