Study of Reinforcement Learning Based Dynamic Traffic Control Mechanism

Zheng Zhang, Seung Jun Baek, Duck Jin Lee and Kil To Chong

Abstract A traffic signal control mechanism is proposed to improve the dynamic response performance of a traffic flow control system in an urban area. The necessary sensor networks are installed in the roads and on the roadside upon which reinforcement learning is adopted as the core algorithm for this mechanism. A traffic policy can be planned online according to the updated situations on the roads based on all the information from the vehicles and the roads. The optimum intersection signals can be learned automatically online. An intersection control system is studied as an example of the mechanism using Q-learning based algorithm and simulation results showed that the proposed mechanism can improve traffic efficiently more than a traditional signaling system.

Keywords Intelligent transportation system · Cooperative vehicle-highway systems · Reinforcement learning · Traffic control mechanism · Intersection signal control

Z. Zhang
Department of Mechanical Engineering, Xian Jiaotong University, Xian, Peoples Republic of China

S. J. Baek · K. T. Chong
Department of Electronics Engineering, Jeonbuk National University, Jeonju, Republic of Korea

D. J. Lee
Department of Mechanical Engineering, Jeonbuk National University, Jeonju, Republic of Korea

K. T. Chong (✉)
Advanced Research Center for Electronics and Information, Jeonbuk National University, Jeonju, Republic of Korea
e-mail: kitchong@chonbuk.ac.kr

J. J. (Jong Hyuk) Park et al. (eds.), *Multimedia and Ubiquitous Engineering*, Lecture Notes in Electrical Engineering 240, DOI: 10.1007/978-94-007-6738-6_129,

1 Introduction

Intelligent Transportation Systems (ITS) utilizes synergistic technologies and systems engineering concepts to develop and improve transportation systems of all kinds [1]. Machine intelligence on the road has been a popular research area with the advent of modern technologies especially artificial intelligence, wireless communication and advanced novel sensors.

Current traffic signal control system design is based on historic traffic flow data which cannot adapt itself to the rapidly varying situations at a crossroad. In some extreme situations, there are no vehicles during a green light and lots of vehicles waiting at a red one.

Many researchers have proposed schemes to solve the afore-mentioned problems like Choy et al. [2] who introduced hybrid agent architecture for real-time signal control. He suggested in his paper a dynamic database for storing all recommendations of the controller agents for each evaluation period. Liu et al. [3] proposed a calculating method of intersection delay under signal control while Bao et al. [4] studied an adaptive traffic signal timing scheme for an isolated intersection. However all these papers solve the problem according to the history flow data but not the current information [5, 6].

This paper makes the following contributions in particular:

(a) A novel traffic flow control mechanism is proposed based on the cooperation of the vehicle, road and traffic management systems. A roadside wireless communication network supports a dynamic traffic flow control method.
(b) Reinforcement learning is introduced as the core algorithm to dynamically plan traffic flow in order to improve efficiency. A Q-learning based intersection traffic signal control system is studied as an example of the proposed mechanism.

2 Study of Intersection Signal Control

In this section, a Q learning algorithm will be used to create a real time cooperation policy for an isolated intersection control under the proposed Traffic Control Mechanism. The algorithm and the simulation are both described in detail. The result shows the advantage of the proposed method.

2.1 Q-Learning Algorithm

Q learning, a type of reinforcement learning, can develop optimal control strategies from delayed rewards, even when an agent has no prior knowledge of the effects of its actions on the environment [7].

The agent's learning task can be described as follows. We require that the agent learn a policy π that maximizes $V^{\pi}(s)$ for all states $\boldsymbol{s}$. We will call such a policy an optimal policy and denote it by π^{*}

$$\pi^{*} \equiv \arg\max_{\pi} V^{\pi}(s), (\forall s) \tag{1}$$

To simplify notation, we will refer to the value function $V^{\pi^{*}}(s)$ of such an optimal policy as $V^{*}(s)$. $V^{*}(s)$ gives the maximum discounted cumulative reward that the agent can obtain starting from state $\boldsymbol{s}$; that is, the discounted cumulative reward obtained by following the optimal policy beginning at state $\boldsymbol{s}$.

However, it is difficult to learn the function $\pi^{*} : S \rightarrow A$ directly, because the available training data does not provide training examples of the form $< s, a >$. Instead, the only training information available to the learner is the sequence of immediate rewards $r(s_i, a_i)$for $i = 0, 1, 2, \ldots$. As we shall see, given this kind of training information it is easier to learn a numerical evaluation function defined over states and actions, then implement the optimal policy in terms of this evaluation function.

What evaluation function should the agent attempt to learn? One obvious choice is V^{*}. The agent should prefer state $\boldsymbol{s}_1$ over state $\boldsymbol{s}_2$ whenever $V^{*}(s_1) > V^{*}(s_2)$, because the cumulative future reward will be greater from $\boldsymbol{s}_1$. The agent's policy must choose among actions, not among states. However, it can use V^{*} in certain settings to choose among actions as well. The optimal action in state $\boldsymbol{s}$ is the action $\boldsymbol{a}$ that maximizes the sum of the immediate reward $r(s, a)$ plus the value V^{*} of the immediate successor state, discounted by γ.

$$\pi^{*}(s) = \arg\max_{a}[r(s, a) + \gamma V^{*}(\delta(s, a))] \tag{2}$$

where $\delta(s, a)$ denotes the state resulting from applying action $\boldsymbol{a}$ to state $\boldsymbol{s}$.

Thus, the agent can acquire the optimal policy by learning V^{*}, provided it has perfect knowledge of the immediate reward function $\boldsymbol{r}$ and the state transition function δ. When the agent knows the functions $\boldsymbol{r}$ and δ used by the environment to respond to its actions, it can then use Eq. (2) to calculate the optimal action for any state $\boldsymbol{s}$.

Unfortunately, learning V^{*} is a useful way to learn the optimal policy only when the agent has perfect knowledge of δ and $\boldsymbol{r}$.

Let us define the evaluation function $Q(s, a)$ so that its value is the maximum discounted cumulative reward that can be achieved starting from state $\boldsymbol{s}$ and applying action $\boldsymbol{a}$ as the first action. In other words, the value of Q is the reward received immediately upon executing action $\boldsymbol{a}$ from state $\boldsymbol{s}$, plus the value (discounted by γ) of following the optimal policy thereafter.

$$Q(s, a) \equiv r(s, a) + \gamma V^{*}(\delta(s, a)) \tag{3}$$

Note that $Q(s, a)$ is exactly the quantity that is maximized in Eq. (3) in order to choose the optimal action $\boldsymbol{a}$ in state $\boldsymbol{s}$. Therefore, we can rewrite Eq. (3) in terms of $Q(s, a)$ as

$$\pi^*(s) = \arg\max_a Q(s, a) \tag{4}$$

Why is this rewrite important? Because it shows that if the agent learns the Q function instead of the V^* function, it will be able to select optimal actions even when it has no knowledge of the functions r and δ. As Eq. (4) makes clear, it need only consider each available action $\boldsymbol{a}$ in its current state $\boldsymbol{s}$ and choose the action that maximizes $Q(s, a)$. This is exactly the most important advantages of Q learning, and also is the reason why we choose Q learning in this paper.

How should the Q learning algorithm be implemented? The key problem is finding a reliable way to estimate training values for Q, given only a sequence of immediate rewards $\boldsymbol{r}$ spread out over time. This can be accomplished through iterative approximation. To see how, notice the close relationship between Q and V^*, $V^*(s) = \max_{a'} Q(s, a')$, which allows rewriting Eq. (3) as follows:

$$Q(s, a) = r(s, a) + \gamma \max_{a'} Q(\delta(s, a), a') \tag{5}$$

Equation (5) provides the basis for algorithms that iteratively approximate Q. In the algorithm, $\overline{Q}$ will be the learner's estimate, or hypothesis of the actual Q function. $\overline{Q}$ will be represented by a large table with a separate entry for each state-action pair. The table can be initially filled with random values (though it is easier to understand the algorithm if one assumes initial values of zero). The agent repeatedly observes its current state $\boldsymbol{s}$, choose some action $\boldsymbol{a}$, executes this action, then observes the resulting reward $r = r(s, a)$ and the new state $s' = \delta(s, a)$. It then updates the table entry for $\overline{Q}(s, a)$ following each such transition, according to the rule:

$$\overline{Q}(s, a) \leftarrow r(s, a) + \gamma \max_{a'} \overline{Q}(s', a') \tag{6}$$

Note that the above training rule uses the agent's current $\overline{Q}$ values for the new state s' to refine its estimate of $\overline{Q}(s, a)$ for the previous state $\boldsymbol{s}$.

The iterative training rule (6) will be replaced by

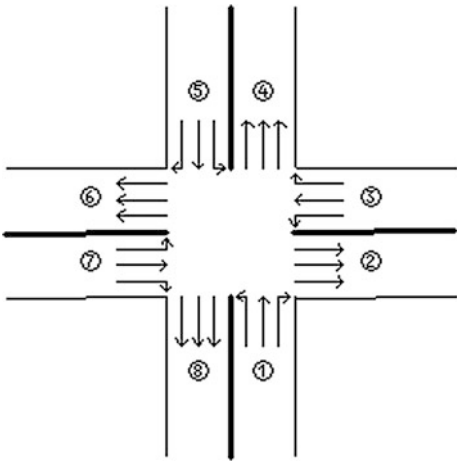
$$\overline{Q}(s, a) \leftarrow g(s, a) + \gamma \min_{a'} \overline{Q}(s', a'). \tag{7}$$

It means that the learning target is to minimize the Q function by minimizing the total cost when acting based on the optimum action sequences. This is exactly the algorithm used in this paper.

2.2 Model of the Intersection Signal System

A traffic system consists of various components, among which the traffic intersection is one of the most important [8]. Our method is applied to a traffic intersection that consists of two intersecting roads, each with several lanes and a set of synchronized traffic lights that manage the flow of vehicles, as shown in Fig. 1.

Fig. 1 Isolated intersection



In this intersection, the rule of traffic management is right-hand based, which is used in China and South Korea. The vehicles in lanes ①, ③, ⑤ and ⑦, are approaching the intersection. Vehicles in ②, ④, ⑥ and ⑧, are leaving the intersection. For each of the approaching lanes, there are three directions for vehicles to choose: turn left, turn right and go straight, as shown in Fig. 1.

We will not consider the turn right direction because it does not impact other directions. In order to make this problem easy to model, we will not consider the pedestrian crossing the road. It will be very easy to add an additional rule for a pedestrian under our proposed mechanism.

Therefore, this problem can be modeled as 8 queues for different paths, as shown in Table 1.

We assume that there are a random number of vehicles spreading on different queues at the beginning of a signal period. This is the initial state of the environment. The final state must be that all the vehicles in the initial state have crossed the intersection. The intersection signal control system is modeled as a leader agent to manage the actions of all vehicle agents around the intersection. Since the action libraries of vehicle agents include actions from A1 to A8, the leader agent can choose any one action or their reasonable combination to reach the final state.

If two of the actions from A1 to A8 are nonintervention, they are possible action combinations. We call these different combinations a signal phase. All possible combinations are shown in Table 2.

Therefore, the problem can be described as how to find the optimum sequence of action combinations to reach the final state. This is the main function of the intersection signal control agent.

For each of the discrete states from the initial state to the final state, the optimum policy will be independent of the previous state. The successor state will be deterministic after one action combination is done. Therefore, this problem can be modeled as a deterministic Markov decision process.

Table 1 Basic action definition of different queues

Queue	Basic action symbol	Path
Que1	A1	① → ④
Que2	A2	① → ⑥
Que3	B1	⑤ → ⑧
Que4	B2	⑤ → ②
Que5	C1	③ → ⑥
Que6	C2	③ → ⑧
Que7	D1	⑦ → ②
Que8	D2	⑦ → ④

Table 2 Action combination symbol

Phase	Action combination symbol	Component
Ph1	Ac1	A1 + A2
Ph2	Ac2	A1 + B1
Ph3	Ac3	B1 + B2
Ph4	Ac4	C1 + C2
Ph5	Ac5	C1 + D1
Ph6	Ac6	D1 + D2

2.3 Parameters of Learning Process

(1) *Cost function*

We suppose that the vehicle number is $\boldsymbol{n}$ at state $\boldsymbol{s}$. After the selected action $\boldsymbol{a}$ completed, the current vehicle number will be $\boldsymbol{n}_1$. The cost of this action depends on the waiting time $\boldsymbol{t}$, and the remainder of vehicles $\boldsymbol{n}_1$.

$$g(s,a) = n_1 \times (t + t_{transition}). \tag{8}$$

where $\boldsymbol{t_{transition}}$ equals one of the three numbers {0, 1.5, 3} shown in Table 3. The average time for each vehicle passing the crossroad is supposed to be 3 s.

Table 3 $t_{transition}$ of different phase transition

Phase transition type	Comment	$t_{transition}$(s)
No transition	Current phase is the same as the previous one	0
Half transition	$Ac1 \Leftrightarrow Ac2; Ac2 \Leftrightarrow Ac3;$ $Ac4 \Leftrightarrow Ac5; Ac5 \Leftrightarrow Ac6;$	1.5
Full transition	Phase transfer except half transition	3

(2) *Discount factor*

In the simulation we set the discount factor, $\gamma = 0.8$.

2.4 Simulation and Results

We wrote some MATLAB code to complete the simulation with the following configuration.

CPU: Intel Pentium 4 Processor 2.40 GHz,
Memory: 1047792 KB,
Operation System: Microsoft Windows XP Professional (SP3).

In order to show the advantage of our proposed mechanism, the traditional signal mechanism was introduced to create a comparative study. In the traditional mechanism, the signal phase transition is in a fixed sequence as shown by Ph1, Ph2, Ph3, Ph4, Ph5 and Ph6. However, our proposed method can determine the optimum phase sequence automatically based on the updated situation.

In the following, we will show the comparative result for three different periods T and different phase time interval t_{phase}.

In the above-mentioned tables, Ps is the simulation period series, NIV is the total number of vehicles at the initial state, Random Queues the number of vehicle queues that are randomly created, TIQ is the time interval from the initial state to the final state for a Q learning method, TWQ is the total waiting time for the Q learning method, TIT is the time interval from the initial state to the final state for the traditional method, $T_{IT} = 6 \times t_{phase}$,

TWT is the total waiting time for the traditional method,

$$P_{EI} = \frac{T_{IT} - T_{IQ}}{T_{IT}} \times 100\ \% \tag{9}$$

Equation (9) determines the percent improvement in the traffic efficiency,

$$P_{WD} = \frac{T_{WT} - T_{WQ}}{T_{WT}} \times 100\ \% \tag{10}$$

Equation (10) shows the percent decrease in total waiting time.

OA is the optimum phase sequence from Q learning, TL is the running time of the Q learning program on the above mentioned computer.

2.5 Analysis of the Results

From Table 4, we find that all the running times of the Q learning program TL in every period are less than one second. This is short enough for the application of the intersection signal control system.

Table 4 Simulation result when $t_{phase} = 60$ s

P_a	N_{IV}	Random queues	T_{IQ} (s)	T_{WQ} (s)	T_{WT} (s)	P_{EI} (%)	P_{WD} (%)	O_A	T_L (s)
1	180	{20 20 40 20 20 20 20 20}	312	18900	24000	13.33	21.25	{4 5 6 1 2 3}	0.8438
2	175	{20 20 19 19 38 19 20 20}	303	17916	23280	15.83	23.04	{1 2 3 6 5 4}	0.9375
3	176	{20 20 18 18 40 20 20 20}	306	18048	23640	15.00	23.65	{1 2 3 6 5 4}	0.4375
4	202	{17 17 36 18 36 18 40 20}	345	28479	30600	4.17	6.93	{1 2 3 6 5 4}	0.8438
5	183	{38 19 16 16 17 17 40 20}	345	21939	26880	4.17	18.38	{3 2 1 4 5 6}	0.8906
6	189	{19 19 36 18 20 20 38 19}	351	23418	28380	2.50	17.48	{1 2 3 4 5 6}	0.9063
7	157	{14 14 16 16 38 19 20 20}	276	14331	22740	23.33	36.98	{3 2 1 6 5 4}	0.8750
8	174	{38 19 26 13 20 20 19 19}	282	18852	22020	21.67	14.39	{4 5 6 1 2 3}	0.9063
9	123	{30 15 14 14 13 13 12 12}	219	8916	14280	39.17	37.35	{4 5 6 3 2 1}	0.8906
10	133	{15 15 17 17 30 15 12 12}	234	10413	16440	35.00	36.66	{3 2 1 6 5 4}	0.9219
11	130	{11 11 34 17 22 11 12 12}	249	11007	16140	30.83	31.80	{6 5 4 1 2 3}	0.8594
12	152	{22 11 15 15 16 16 38 19}	285	14904	24480	20.83	39.12	{3 2 1 4 5 6}	0.8750
13	171	{36 18 32 16 24 12 22 11}	273	19292	19500	24.17	1.07	{4 5 6 1 2 3}	0.9531
14	183	{26 13 36 18 26 13 34 17}	300	22265	25560	16.67	12.89	{6 5 4 3 2 1}	0.9375
15	120	{32 16 20 10 20 10 6 6}	216	10221	11760	40.00	13.09	{6 5 4 1 2 3}	0.8750
16	128	{19 19 20 10 15 15 15 15}	219	9768	15900	39.17	38.57	{4 5 6 1 2 3}	0.8906
17	112	{14 14 9 9 28 14 16 8}	189	7905	14220	47.50	44.41	{1 2 3 4 5 6}	0.8906
18	100	{16 8 8 8 34 17 6 3}	195	6480	11760	45.83	44.90	{3 2 1 6 5 4}	0.9375
19	128	{16 16 32 16 16 8 16 8}	228	9960	15120	36.67	34.13	{4 5 6 1 2 3}	0.8906

(continued)

Table 4 (continued)

P_a	N_{IV}	Random queues	T_{IQ} (s)	T_{WQ} (s)	T_{WT} (s)	P_{EI} (%)	P_{WD} (%)	O_A	T_L (s)
20	128	{15 15 30 15 147 16 16}	237	10431	16620	34.17	37.24	{6 5 4 1 2 3}	0.8906
21	108	{16 8 36 18 14 14 1 1}	189	6906	10860	47.50	36.41	{4 5 6 3 2 1}	0.8750
22	81	{5 5 24 12 18 9 4 4}	165	4365	10140	54.17	56.95	{6 5 4 1 2 3}	0.4844
23	109	{14 14 18 9 30 15 6 3}	207	7740	11400	42.50	32.11	{1 2 3 6 5 4}	0.4375
24	144	{9 9 16 16 40 20 17 17}	258	11772	21660	28.33	45.65	{3 2 1 6 5 4}	0.5156
25	77	{5 5 30 15 8 4 5 5}	156	3501	9060	56.67	61.36	{6 5 4 1 2 3}	0.4219

At the same time, the percent traffic efficiency improvement PEI, is located in [4.17 % 47.5 %], the percent total waiting time decrease PWD is located in [1.07 % 56.95 %]. The average percents of PEI are 32.2 % and the average percents of PWD are 37.5 %.

3 Conclusion

A new traffic control based mechanism based on a combination of machine learning and multiagent modeling methods is proposed for future intelligent transportation systems. The control systems, the vehicles, and some necessary roadside sensors are all modeled as intelligent agents in the proposed systems, therefore the ITS system will be a multiagent system. It is possible to improve the traffic control efficiency by some artificial intelligence algorithm.

The control method for an isolated intersection was studied specifically. The intersection signal was first modeled according to the proposed mechanism then a new algorithm based on reinforcement learning, especially Q-learning, was proposed and studied in detail. A simulation for such an intersection system was finally carried out and a comparative study with the traditional intersectional signal method was done.

Simulation results showed that the proposed intersection control mechanism can improve traffic efficiency by more than 30 % over the traditional method and simultaneously bring the drivers some benefit by decreasing the waiting time by more than 30 %. This proves that the proposed traffic control mechanism is applicable in the near future.

Acknowledgments This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2012-038978) and (No. 2012-0002434).

References

1. http://www.ewh.ieee.org/tc/its/
2. Choy MC, Srinivasan D, Cheu RL (2003) Cooperative, hybrid agent architecture for real-time traffic signal control. IEEE Trans Syst Man Cybern Part A Syst Hum 33(5):597–607
3. Liu G, Zhai R, Pei Y (2007) A calculating method of intersection delay under signal control. In: Proceedings of the 2007 IEEE intelligent transportation systems conference, Seattle, pp 1114–1119
4. Bao W, Chen Q, Xu X (2006) An adaptive traffic signal timing scheme for bus priority at isolated intersection. In: Proceedings of the 6th world congress on intelligent control and automation, Dalian, pp 8712–8716
5. Srinivasan D, Choy MC (2006) Cooperative multi-agent system for coordinated traffic signal control. IEE Proc Intell Transp Syst 153(1):41–50
6. Lee JH, Lee-Kwang H (1999) Distributed and cooperative fuzzy controllers for traffic intersections group. IEEE Trans Syst Man Cybern C Appl Rev 29:263–271
7. Mitchell TM (1997) Machine learning. McGraw-Hill, New York. ISBN: 0070428077
8. D'Ambrogio A et al (2008) Simulation model building of traffic intersections. Simul Model Pract Theory