Research Article

The Study of Reinforcement Learning for Traffic Self-Adaptive Control under Multiagent Markov Game Environment

Lun-Hui Xu,^{1,2} Xin-Hai Xia,^{1,3} and Qiang Luo¹

¹ School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China

² School of Electronic Engineering and Automation, Jiangxi University of Science and Technology, Ganzhou, Jiangxi 341000, China

³ School of Port and Shipping Management, Guangzhou Marine Institute, Guangzhou 510725, China

Correspondence should be addressed to Xin-Hai Xia; xiaxinhai@126.com

Received 25 February 2013; Revised 12 August 2013; Accepted 26 August 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Lun-Hui Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Urban traffic self-adaptive control problem is dynamic and uncertain, so the states of traffic environment are hard to be observed. Efficient agent which controls a single intersection can be discovered automatically via multiagent reinforcement learning. However, in the majority of the previous works on this approach, each agent needed perfect observed information when interacting with the environment and learned individually with less efficient coordination. This study casts traffic self-adaptive control as a multiagent Markov game problem. The design employs traffic signal control agent (TSCA) for each signalized intersection that coordinates with neighboring TSCAs. A mathematical model for TSCAs' interaction is built based on nonzero-sum markov game which has been applied to let TSCAs learn how to cooperate. A multiagent Markov game reinforcement learning approach is constructed on the basis of single-agent Q-learning. This method lets each TSCA learn to update its Q-values under the joint actions and imperfect information. The convergence of the proposed algorithm is analyzed theoretically. The simulation results show that the proposed method is convergent and effective in realistic traffic self-adaptive control setting.

1. Introduction

As car ownership rates and traffic volume have steadily increased over the last decades, existing road infrastructure today is often strained nearly to its limits. Continuous expansion of this infrastructure, however, is not possible or even desirable due to spatial, economic, and environmental reasons. It is therefore of paramount importance to try to optimize the flow of traffic in a given infrastructure. Traffic self-adaptive control of multiple intersections is synergetic and has the potential to significantly alleviate traffic congestion in urban transportation networks as opposed to the commonly used fixed timing and actuated control systems. Existing traditional traffic adaptive control systems such as TRANSYT, SCOOT, SCATS, and sophisticated dynamic programming approach [1] do not have a mechanism for learning from feedback on the quality of their model, which may lead to systematic errors.

Several researchers have employed classical control methods such as fuzzy logic [2], neural networks [3], and evolutionary algorithms [4] to traffic self-adaptive control. These

methods perform well but cannot be adapted to the changing characteristics of traffic flow. Reinforcement learning (RL) [5] is able to perpetually learn and improve the service over time. Multiagent reinforcement learning is an extension of RL to multiple agents in stochastic environment. The decentralized traffic control problem is an excellent test for multiagent reinforcement learning due to the inherited dynamics and stochastic nature of the traffic system [6–9].

There are two shortages in the application of multiagent reinforcement learning to traffic self-adaptive control problem as discussed below.

- (1) *With Less Efficient Coordination.* The majority of the previous studies consider independent learning agents which do not include any explicit mechanism for coordination [6–14]. Only a few previous studies consider coordination mechanism between the learning agents. Kuyer et al. [15] consider explicit two-level coordination mechanism between the learning agents that extends Wiering [6] using the coordination graphs. Max-plus algorithm is used to estimate the

FIGURE 1: The structure model of TSCA.

optimal joint action by sending locally optimized messages among connected agents. However, Max-plus algorithm is computationally demanding, and therefore the agents report their current best action at any time even if the action found so far may be sub-optimal. Tantawy and Abdulhai [16] presented multi-agent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC) which maintains a coordination mechanism (indirect coordination and direct coordination) between agents without compromising the dimensionality of the problem. Indirect coordination is realized by best-response multiagent learning in nonstationary environments, and the direct coordination is typically based on communication.

- (2) *On the Assumption of the Complete Knowledge.* In addition, since the traffic environment is changing in time and intersection cannot fully understand other intersections' information such as traffic arrival rate, vehicle queue, or delays, it is over idealized that the utility matrix of each agent is public; that is, perfect observed information is required when agents are interacting with the environment. According to this assumption, agents may select individual actions that are locally optimal but that together result in global inefficiencies. So this assumption is not too realistic.

It is argued that the use of a model-based RL approach adds unnecessary complexities compared with using model-free Q-learning. To overcome the upper deficiencies of the previous approach, this paper conducts a multiagent Markov game reinforcement learning method for optimizing traffic self-adaptive control. We define a TSCA for each signalized intersection that coordinates with neighboring agents. A

mathematical model for TSCAs' interaction is built based on nonzero-sum Markov game which has been applied to let TSCAs learn how to cooperate. A multiagent Markov game reinforcement learning approach is constructed on the basis of single-agent Q-learning. This method let each TSCA learn to update its Q-values under the joint actions and imperfect information. Convergence and effectiveness of the improving algorithm are verified.

2. Structure Model for TSCA

We defined a TSCA for each urban signalized intersection that takes the charge of controlling all signal phases. Its main function is to establish corresponding control strategy which will be implemented by signal light according to the current traffic state of both of its own and of its neighbors [14]. Therefore, the intersection's traffic flow conditions are improved. Figure 1 shows the structure model for TSCA. The reinforcement signal is a reward function r^i which will be defined in Section 3.

The structure model of TSCA has been shown in Figure 1. As we can see in Figure 2, it is mainly composed of learning module, action decision making module, communication module, and coordination module. Learning module infers whether there are reasonable regulations from real-time observational data. If some reasonable regulations exist, learning module executes these regulations and determines signal control plan. Coordination module analyzes the present traffic state of TSCA to decide if it is necessary to send messages to the adjacent TSCAs, and it deals with the TSCA coordination. Communication module is mostly responsible for the communication with the adjacent TSCAs. Action decision making module resolves function of reasoning and decision making of TSCA.

FIGURE 2: Traffic network used in the simulations.

In most cases, average delay time is sufficient to determine intersection's relative traffic performance; that is, lower average delay time implies lower ratio of stopped vehicles and total queue length. Vehicle's delay of coordinated controlled intersection is composed of normal delay, random delay, and oversaturation delay which is the same as isolated controlled ones. The computation of normal delay needs vehicle arriving and leaving graph. Since every incoming traffic flow of an intersection is up to green light time and released ratio of upstream intersections, arrival rate is not a constant but a time-varying functional expression. For random over saturation delay, the degree of traffic flow's random fluctuation within each cycle in coordinated control is far away less than that in isolated control, so delay value will decrease. Intersection's delay time can adopt the following transient functional model [17]:

$$\bar{d} = \begin{cases} \frac{C(1-\lambda)^2}{2(1-\gamma)} + \frac{N_0x}{q}, & x < 1, \\ \frac{R}{2} + \frac{N_0x}{q}, & x \geq 1, \end{cases} \quad (1)$$

$$N_0 = \begin{cases} \frac{q_c T}{4} \left[(x-1) + \sqrt{(x-1)^2 + \frac{6(x-x_0)}{q_c T}} \right], & x > x_0, \\ 0, & x \leq x_0, \end{cases} \quad (2)$$

where $\bar{d}$ is the average delay time per vehicle (s/pcu); C is a signal cycle length (s); λ is a split; γ is a ratio of flow volume; N_0 is the average length of over saturation stopped fleet (pcu); q_c is the intersection's traffic capacity (pcu/h); T is a time interval (h); x is the intersection's saturation; q is the vehicle's arrival rate (pcu/h); and R is the length of red light time.

3. Mathematical Model for TSCAs' Interaction Based on Nonzero-Sum Markov Game

Game theory is the best mathematical tool to study human society's interaction. The interaction between TSCAs comes down to game model [18]. In the dynamic traffic signal

control system of multiple intersections, each intersection's signal timing scheme not only affects directly its neighbor's traffic but also indirectly affects nonneighboring intersection's traffic. Although, in traffic networks, the TSCA is incapable of observing the conditions of the entire network, it is possible to observe the conditions of the neighboring TSCAs. In order to avoid the excessive complexity and frequency of TSCA's interaction brought by the increase in the number of controlled intersections, we limit that every TSCA to only interact with the adjacent one. Since two adjacent TSCAs can form a coalition to acquire a whole and local optimal performance in view of the openness of bilateral information, the interaction between two adjacent TSCAs conforms to a two-matrix nonzero-sum cooperative game [19].

Markov decision process (MDP) is widely researched which represents the problem about single agent in multiple states. By contrast, two matrix games were used to solve the problem for multiple agents in single state. Markov game can be regarded as the combination of MDP and two-matrix game which defines the frame of multiple agents and multiple environment states [18–20]. Since the traffic environments which TSCA is confronted with have the characteristics of dynamic, complexity, uncertainty, and openness, n -player nonzero-sum Markov game is suitable to be used to establish the interaction model for TSCAs.

An n -player nonzero-sum Markov game can be defined by a tuple $M = \langle N, S, A, p, r \rangle$, where the parameters N, S, A, p, r are explained in detail as follows according to traffic self-adaptive control.

N is a set of finite interactive TSCAs.

$S = S^1 \times S^2 \times \dots \times S^i \times \dots \times S^N$ is a set of finite states. Let s^i be a local state of TSCA i ; $s = (s^1, s^2, \dots, s^n)$ is a global state which is decomposed into local states. s^i is defined by a vector of two components. The first component is the position for the first vehicle approaching TSCA i from the directions of west, north, east, and south. Since the size of the state space grows rapidly if the accurate distance from the vehicle to TSCA represents local state, all lanes connected to TSCA are divided into a certain number of equal sections. The sections are sequentially encoded from the vehicle nearest to TSCA. So, the section's code can be used to define the local state s^i . The second component is the maximum queue lengths associated with each direction defined as follows [16]:

$$\max_{l \in L_i} q_l^k, \quad (3)$$

where q_l^k is the number of queued vehicles in lane l at time k . A vehicle is considered at a queue if its speed is below a certain speed threshold (Sp_{thr}). q_l^k is computed as follows:

$$q_l^k = q_l^{k-1} + \sum_{v \in V_l^k} q_v^k,$$

$$q_v^k = \begin{cases} 1, & \text{if } Sp_v^{k-1} > Sp_{thr} \text{ and } Sp_v^k \leq Sp_{thr}, \\ -1, & \text{if } Sp_v^{k-1} \leq Sp_{thr} \text{ and } Sp_v^k > Sp_{thr}, \\ 0, & \text{if } Sp_v^{k-1} \leq Sp_{thr} \text{ and } Sp_v^k \leq Sp_{thr}, \end{cases} \quad (4)$$

where V_l^k is the set of vehicles traveling on lane l at time k .

$A = A^1 \times A^2 \times \dots \times A^i \times \dots \times A^N$ is a set of joint signal timing actions for multiple TSCAs, where A^i is the subset of finite signal timing actions for TSCA i . a^i represents the current phase duration which depends on intersection's phase and the adjustment of green time for the current green phase, $a^i \in A^i$. The adjustment of green time is to be determined by the relation of section's length and vehicle's velocity, for example, {green time plus 1 s, green light time plus 2 s, green time minus 1 s, green time minus 2 s, unchangeably}. Phase timing signal scheme generically consists of east-west straight and right turn, south-north straight and right turn, east-west left turn, and south-north left turn. a represents a joint signal timing action for the n TSCAs, $a \in A$.

$p : S \times A \rightarrow [0, 1]$ is a state transition function mapping a present state s and a joint signal timing action a to a probability over states.

$r = \{r^1, r^2, \dots, r^i, \dots, r^N\}$ is a set of reward functions for all TSCAs, where $r^i : S \times A \rightarrow R$ is a reward function for TSCA i mapping state-action tuples to immediate scalar rewards. $r^i(s, a)$ represents a reward function for TSCA i when the TSCAs take the joint action a in the state s . $r^i(s, a)$ can be expressed by the division value between total volume for passing vehicles and accumulative waiting time.

Suppose that the traffic flow is random and corresponds to Poisson distribution. When green light of TSCA i in south-north direction is open, $r^i(s, a)$ is described in a mathematical equation as follows:

$$r^i(s, a) = \frac{\delta_n(a^i - t_n) + \delta_s(a^i - t_s)}{(1/2)k\delta_w(a^i - t_w)^2 - (1/2)k\delta_e(a^i - t_e)^2}. \quad (5)$$

When green light of TSCA i in east-west direction is open, $r^i(s, a)$ is computed as follows:

$$r^i(s, a) = \frac{\delta_w(a^i - t_w) + \delta_e(a^i - t_e)}{(1/2)k\delta_n(a^i - t_n)^2 - (1/2)k\delta_s(a^i - t_s)^2}, \quad (6)$$

where a^i is a signal timing action for TSCA i and t_w, t_n, t_e , and t_s represent the time to reach TSCA i for the first vehicle incoming but not passing TSCA i from the directions of west, north, east, and south, respectively, beginning with the alternation of red and green light. If $a^i < t$ (t represents t_w, t_n, t_e , or t_s), we set $a^i - t = 0$. $\delta_w, \delta_n, \delta_e$, and δ_s represent the intensity for Poisson flow of vehicles incoming TSCA i from the directions of west, north, east, and south, respectively. δ depends on the state and timing signal action of TSCAs adjacent to TSCA i . It can generalize from the average value of statistical historical data. k is a positive constant which represents the degree of punishment.

Let π^i be the probability distributions over action set A^i of TSCA i . With each $s \in S$, there is an n -player game $\Gamma = \{(\pi^1, \pi^2, \dots, \pi^n)\}$. $V^i(s, \pi^1, \pi^2, \dots, \pi^n)$ is TSCA i 's total discounted reward in state s and under joint strategy $(\pi^1, \pi^2, \dots, \pi^n)$. Suppose that $\Pi = \{\pi^i, i \in N\}$ is a joint strategy when each TSCA selects action a^i with a probability π^i . We define $\Pi^{-i} = \{\pi^j, j \neq i, i, j \in N\}$; for each given

Π^{-i} , TSCA i chooses a corresponding optimal strategy $\pi^{*i} = \text{argmax} V\{\Pi^{-i} \cup \pi^i\}$.

In an n -player nonzero-sum Markov game under mixed strategy, a Nash equilibrium point is a tuple of n strategies $(\pi^{*1}, \dots, \pi^{*i}, \dots, \pi^{*n})$ such that, for all $s \in S$ and $i = 1, \dots, n$,

$$V^i(\pi^{*1}, \dots, \pi^i, \dots, \pi^{*n}) \leq V^i(\pi^{*1}, \dots, \pi^{*i}, \dots, \pi^{*n}), \quad (7)$$

$$\forall \pi^i \in \Delta(A^i),$$

where $\Delta(A^i)$ is the space of probability distributions over the TSCA i 's actions.

In the nonzero-sum Markov game, TSCA i can gain more rewards via cooperation than that under independent action. A Nash equilibrium point can be reached when none TSCAs gain more optimal policy.

4. Multiagent Markov Game Reinforcement for Traffic Self-Adaptive Control

4.1. Single-Agent Q-Learning. Q-learning which was presented by Watkins defines a learning method within a Markov decision process [21]. The basic idea of Q-learning is that we can define a function Q such that

$$Q^*(s, a) = r(s, a) + \gamma \sum_{s'} p(s' | s, a) V(s', \pi^*), \quad (8)$$

$$V(s, \pi^*) = \max_a Q^*(s, a). \quad (9)$$

By this definition, $\gamma \in [0, 1]$ is a discount factor and is used to discount future rewards. $p(s' | s, a)$ is the probability of transitioning to state s' after taking action a in state s . A solution π^* that satisfies (9) is guaranteed to be an optimal policy. $Q^*(s, a)$ is the total discounted reward of taking action a in state s and then following the optimal policy thereafter.

If $Q^*(s, a)$ is given, then the optimal policy π^* can be found by simply identifying the action that maximizes $Q^*(s, a)$ under the state s . The problem is then reduced to finding the function $Q^*(s, a)$ instead of searching for the optimal value of $V(s, \pi^*)$.

Q-learning provides us with a simple updating procedure, in which the TSCA starts with arbitrary initial values of $Q(s, a)$ for all $s \in S, a \in A$ and updates the Q -values as follows:

$$Q_{t+1}(s, a) = (1 - \alpha) Q_t(s, a) + \alpha \left[r_t + \gamma \max_b Q_t(s', b) \right], \quad (10)$$

where $\alpha \in [0, 1]$ is the learning rate sequence.

Watkins and Dayan proved that sequence (8) converges to $Q^*(s, a)$ under the assumption that all states and actions have been visited infinitely often and the learning rate satisfies certain constraints [21].

Even in single-agent Q-learning approach that is proven to optimally converge to the joint policy, each TSCA has to keep a set of tables whose size is exponential in the number of agents: $|A_1| \times \dots \times |A_n| \times |S_1| \times \dots \times |S_n|$. In addition to the dimensionality issue, the method requires each TSCA to observe

the state of the whole system which is infeasible in the case of transportation networks. Single-agent Q-learning takes other TSCAs as a part of environment and it updates future rewards based on merely the TSCA's own maximum payoff regardless of other TSCAs' actions. In MAS, we adopted a multiagent Markov game reinforcement method in which each TSCA updates its $Q(s, a)$ according to immediate reward by interacting with other TSCAs and observing actions taken by all other TSCAs and others' rewards.

4.2. Multiagent Markov Game Reinforcement Learning for Traffic Self-Adaptive Control. In this method, at each time t , TSCA k updates its own Q-values Q_t^k and learn about other adjacent TSCAs' Q-values Q_t^j ($j \in \{1, \dots, n\}, j \neq k$), by observing its own reward, actions taken by neighboring TSCAs, neighbor's rewards, and the new state s' . In the state s_t , TSCA k calculates a Nash equilibrium $\pi^1(s_t), \dots, \pi^k(s_t), \dots, \pi^n(s_t)$ for the stage game $(Q_t^1(s_t), \dots, Q_t^k(s_t), \dots, Q_t^n(s_t))$. Then, TSCA k 's value function $V_t^k(s_t)$ is its payoff in state s_t for the selected equilibrium which can be computed as follows:

$$\begin{aligned} V_t^k(s_t) &= \text{Nash } Q_t^k(s_t) \\ &= \pi^1(s_t) \cdots \pi^k(s_t) \cdots \pi^n(s_t) Q_t^k(s_t). \end{aligned} \quad (11)$$

TSCA k updates its Q-values according to

$$\begin{aligned} Q_{t+1}^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^k + \gamma \text{Nash } Q_t^k(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^k + \gamma \pi^1(s_{t+1}) \cdots \pi^k(s_{t+1}) \cdots \\ &\quad \quad \pi^n(s_{t+1}) Q_t^k(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t \left[r_t^k + \gamma \prod_i \pi^i(s_{t+1}) Q_t^k(s_{t+1}) \right]. \end{aligned} \quad (12)$$

Information about other TSCAs' Q-values is not given, so TSCA k must learn about them too. TSCA k updates the beliefs about TSCA j 's Q-function according to the same rule (12) which is applied to its own,

$$\begin{aligned} Q_{t+1}^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^j + \gamma \text{Nash } Q_t^j(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \end{aligned}$$

$$\begin{aligned} &+ \alpha_t [r_t^j + \gamma \pi^1(s_{t+1}) \cdots \pi^k(s_{t+1}) \cdots \\ &\quad \pi^n(s_{t+1}) Q_t^j(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t \left[r_t^j + \gamma \prod_i \pi^i(s_{t+1}) Q_t^j(s_{t+1}) \right]. \end{aligned} \quad (13)$$

Note that $(\pi^1, \dots, \pi^k, \dots, \pi^n)$ is a joint mixed strategy to a Nash equilibrium point. r_t^k and s_{t+1} are TSCA k 's observed information. In order to update Q-values, TSCA k would know the priori knowledge about other TSCAs' strategy value π^i ($i \neq k$). That is to say, we should solve a set composed of (13) when $i = 1, \dots, k-1, k+1, \dots, n$. This is an n-order nonlinear problem which has no practical solutions. In addition, since the TSCA's observed information is imperfect, and in order to avoid the curse of dimensionality, we use probability statistics and Bayes method to estimate beliefs about other TSCAs' mixed strategies. And in such coordination mechanism, TSCA can reach a unique equilibrium.

Let us define that TSCA j conjecture TSCA i takes the action a^i in the probability of $p^k(a^i)$. The probability $p^k(a^i)$ can be calculated as follows via Boltzmann formula:

$$p^k(a^i) = \frac{e^{Q(s_t, a^i)/T}}{\sum_{a \in A^i} e^{Q(s_t, a)/T}}, \quad (14)$$

where T is a temperature parameter which reflects exploring degree and decreases with time.

Each TSCA takes its own actions in state s_t ; then TSCA k observes other TSCAs' taken actions and new state s_{t+1} ; after that, TSCA k updates the belief about other TSCAs' actions. According to Bayes formula, the belief about TSCA j 's actions can be computed as follows:

$$p(a^i | a^k, s_{t+1}) = \frac{p(s_{t+1} | a^i, a^k) p(a^i)}{p(s_{t+1} | a^k)}, \quad (15)$$

where $p(s_{t+1} | a^i, a^k)$ is the probability of transiting to state s_{t+1} after TSCA i and TSCA k take joint actions in state s_t and $p(s_{t+1} | a^k)$ is the probability of transiting to state s_{t+1} after TSCA k takes its own action independently in state s_t . The probability of $p(s_{t+1} | a^k)$ and $p(s_{t+1} | a^i, a^k)$ can be acquired from environment knowledge. $p(a^i)$ represents the probability in which TSCA i will take the action a^i . $p^k(a^i)$ can similarly replace the estimate of $p^k(a^i)$.

Furthermore, TSCA k 's belief about other multiple TSCAs' joint actions can be generalized from (15) as follows:

$$p(a^1, \dots, a^n | a^k, s_{t+1}) = \prod_{i \neq k} p(a^i | a^k, s_{t+1}). \quad (16)$$

According to the analysis above, the multiagent Markov game reinforcement algorithm is summarized as follows by taking TSCA k for example.

Initialize

- (1) Let $t = 0$; get the initial state s_0 . For any $s \in S$ and any $a^k \in A^k$, let $\pi^k(s, a^k) = 1/|A|$ and $Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) = 0$.

Loop

- (2) Choose action a_t^k : TSCA k chooses the action a_t^k by the probability of $\pi^k(s, a^k)$ which is the best-response mixed policy of the matrix game $Q_t^k(s, a^1, \dots, a^n)$ under the joint actions of other TSCAs.
- (3) Observe $r_t^1, \dots, r_t^n; a_t^1, \dots, a_t^n$; and $s_{t+1} = s'$.
- (4) TSCA k updates the belief about other TSCAs' actions according to Bayes formula (15).
- (5) Update Q_{t+1}^k for $k = 1, \dots, n$. Consider

$$\begin{aligned} Q_{t+1}^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ = (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ + \alpha_t [r_t^k + \gamma p(a^1, \dots, a^n | a^k, s_{t+1}) Q_t^k(s_{t+1})], \end{aligned} \quad (17)$$

where $p(a^1, \dots, a^n | a^k, s_{t+1})$ is defined in (16).

- (6) Let $t = t + 1$ return to (2).

5. Convergence Proof

Assumption 1. For any multiagent game $(Q^1(s), \dots, Q^N(s))$, its Nash equilibrium point $(\pi^1, \pi^2, \dots, \pi^N)$ has the following properties.

- (i) If the Nash equilibrium does not reach a global optimum, then the TSCA which takes the policy of Nash equilibrium will get more payoffs when the other TSCAs' policies deviate from Nash equilibrium:

$$\begin{aligned} \prod_{j \in N} \pi^j(s) \cdot \sum_{i \in N} Q^i(s) &\leq \sum_{i \in N} Q^i(s) \cdot \pi^i(s) \cdot \prod_{j \in N, j \neq i} \hat{\pi}^j(s), \\ \forall \hat{\pi}^j(s) &\in \Delta(A^j). \end{aligned} \quad (18)$$

- (ii) If the Nash equilibrium reaches a global optimum, then

$$\begin{aligned} \prod_{j \in N} \pi^j(s) \cdot \sum_{i \in N} Q^i(s) &\geq \sum_{i \in N} Q^i(s) \cdot \prod_{j \in N} \hat{\pi}^j(s), \\ \forall \hat{\pi}^j(s) &\in \Delta(A^j). \end{aligned} \quad (19)$$

Assumption 2. Learning sequence $\{\alpha_t\}$ satisfies the following:

- (i) $0 \leq \alpha_t \leq 1$, $\sum_{t=1}^{\infty} \alpha_t = \infty$, $\sum_{t=1}^{\infty} \alpha_t^2 < C < \infty$, and the latter two hold uniformly and with probability 1, where C is constant,

- (ii) if $(s, a^i, a^{-i}) \neq (s_t, a_t^i, a_t^{-i})$, then $\alpha_t(s, a^i, a^{-i}) = 0$, where $i \in N$. Under Assumption (i), given a finite MDP, the Q-learning algorithm proposed by [21] converges with probability 1 to the optimal Q-function. The second item in Assumption 2 states that the agent updates only the Q-function element corresponding to current state s_t and actions $a_t^1, \dots, a_t^n$.

Lemma 3 (see [22]). Under Assumption 2, iterative process $Q_{t+1} = (1 - \alpha_t)Q_t + \alpha_t w_t$ will converge to $E(w_t | h_t, \alpha_t)$, with probability 1, where h_t is the historical states and policies at the time t .

Lemma 4 (see [23]). $\forall s \in S$, assume that $(\pi^1(s), \pi^2(s), \dots, \pi^N(s))$ is the Nash equilibrium point for the stage game $(Q^1(s), Q^2(s), \dots, Q^N(s))$, its Nash equilibrium payoff is $(v^1(s, \pi^1, \pi^{-1}), v^1(s, \pi^2, \pi^{-2}), \dots, v^1(s, \pi^N, \pi^{-N}))$, then the TSCA i 's Nash equilibrium payoff under joint actions is

$$\begin{aligned} Q^i(s, a^1, \dots, a^N) &= r(s, a^1, \dots, a^N) \\ &+ \gamma \sum_{s'=1}^N p(s' | s, a^1, \dots, a^N) \\ &\times v^i(s', \pi^1, \pi^2, \dots, \pi^N). \end{aligned} \quad (20)$$

Lemma 5 (see [23]). Given that $P_t^i Q^i(s) = r_t^i + \gamma Q^i(s) \prod_{j \in N} \pi^j(s)$ where, $\forall s \in S$, $(\pi^1(s), \pi^2(s), \dots, \pi^N(s))$ is the mixed Nash equilibrium policy for the stage game $(Q^1(s), Q^2(s), \dots, Q^N(s))$, then $P_t = (P_t^1, \dots, P_t^N)$ is a contract mapping.

Lemma 6 (see [22]). Under Assumption 2, if the iterative process $U_{t+1}(x) = (1 - \alpha_t(x))U_t(x) + \alpha_t(x)[P_t v^*](x)$ holds and P_t satisfies $\|P_t V - P_t v^*\| \leq \gamma \|V - v^*\| + \lambda_t$, $\forall V$, where $0 \leq \gamma \leq 1$, $\lambda_t \geq 0$ and converges to 0 with probability 1, then the following iterative process will converge to v^* with probability 1:

$$V_{t+1}(x) = (1 - \alpha_t(x)) V_t(x) + \alpha_t(x) [P_t v_t](x). \quad (21)$$

Under the above assumptions and lemmas, if the Nash equilibrium point for the stage game is obtained, then proposed method converges to game Q-values of equilibrium point with probability 1.

Theorem 7. Assume that the game sequence $(Q_t^1(s), \dots, Q_t^N(s))$ for each $s \in S(Q_t^1(s), \dots, Q_t^N(s))$ satisfies

$$\begin{aligned} Q_{t+1}^i(s, a^1, \dots, a^N) &= (1 - \alpha_t) Q_t^i(s, a^1, \dots, a^N) \\ &+ \alpha_t \left[r_t^i + \gamma Q_t^i(s') \prod_{j \in N} \pi^j(s') \right], \end{aligned} \quad (22)$$

then the game sequence converges to Q -value of equilibrium point $(Q^{1*}, \dots, Q^{N*})$ with probability 1, where the Q -value meets

$$\begin{aligned} Q^{i*}(s, a^1, \dots, a^N) &= r^i(s, a^1, \dots, a^N) \\ &+ \gamma \sum_{s'=1}^N p(s' | s, a^1, \dots, a^N) \\ &\quad \times v^i(s', \pi^{1*}, \pi^{2*}, \dots, \pi^{N*}), \end{aligned} \quad (23)$$

where $i \in N$; $(\pi^1(s'), \pi^2(s'), \dots, \pi^N(s'))$ is the mixed Nash equilibrium policy for the stage game $(Q_t^1(s'), \dots, Q_t^N(s'))$ at the state s' ; $(\pi^{1*}, \pi^{2*}, \dots, \pi^{N*})$ is the Nash equilibrium point for the game.

Proof. Under Lemma 4,

$$\|p_t^i Q^i - p_t^i Q^{i*}\| \leq \gamma \|Q^i - Q^{i*}\|. \quad (24)$$

Under Lemma 3,

$$\begin{aligned} Q_{t+1}^k(s, a^1, \dots, a^N) &= (1 - a_t) Q_t^k(s, a^1, \dots, a^N) \\ &+ a_t \left[r_t^i + \gamma Q_t^i(s') \prod_{j \in N} \pi^j(s') \right] \end{aligned} \quad (25)$$

converges to

$$\begin{aligned} &E \left(r_t^i + \gamma Q^i(s) \prod_{j \in N} \pi^j(s) \right) \\ &= \sum_{s'} p(s' | s, a^1, \dots, a^N) \\ &\quad \cdot \left(r_t^i(s, a^1, \dots, a^N) + \gamma Q^i(s') \prod_{j \in N} \pi^j(s') \right). \end{aligned} \quad (26)$$

Defining mapping T^i as

$$\begin{aligned} (T^i Q^i)(s' | s, a^1, \dots, a^N) \\ &= \sum_{s'} p(s' | s, a^1, \dots, a^N) \\ &\quad \cdot \left(r_t^i(s, a^1, \dots, a^N) + \gamma Q^i(s') \prod_{j \in N} \pi^j(s') \right), \end{aligned} \quad (27)$$

then $T^i Q^i = \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot P_t Q^i(s)$. Under Lemma 5, since P_t is the contract mapping of Q^i and $p(s' | s, a^1, \dots, a^N) \geq 0$, T^i is also the contract mapping of Q^i and Q^{i*} is the fixed point of mapping T^i . Consider

$$\begin{aligned} (T^i Q^{i*})(s, a^1, \dots, a^N) \\ &= \sum_{s'} p(s' | s, a^1, \dots, a^N) \end{aligned}$$

$$\begin{aligned} &\cdot \left(r_t^i(s, a^1, \dots, a^N) + \gamma Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') \right) \\ &= r_t^i(s, a^1, \dots, a^N) \\ &+ \gamma \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s'). \end{aligned} \quad (28)$$

Under Lemma 4, $Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') = v^i(s', \pi^{1*}, \dots, \pi^{N*})$, so $Q^{i*} = T^i Q^i$. Therefore the iterative formula

$$\begin{aligned} Q_{t+1}^k(s, a^1, \dots, a^N) &= (1 - \alpha) Q_t^k(s, a^1, \dots, a^N) \\ &+ \alpha \left[r_t^i + \gamma Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') \right] \end{aligned} \quad (29)$$

converges to $T^i Q^i = Q^{i*}$ with probability 1. Under Lemma 6, formula (20) converges to Q^{i*} . $\square$

6. Simulation

6.1. Analysis of the Method's Convergence. We consider the traffic network shown in Figure 2 used for the scenery to test the proposed approach described in Section 4. Paramics, a microscopic traffic simulator, is used to build the testbed network. The multiagent Markov game reinforcement learning algorithm is written in Matlab as a stand-alone application. The interaction between the multiagent Markov game reinforcement learning algorithm and the Paramics environment is implemented through the application programming interface (API) functions in Paramics.

δ , defined as the intensity of Poisson flow observed entering the traffic network, obeys uniform distribution in the interval of [5, 18]. The traffic network includes 5 intersections in which each direction has four lanes. Each intersection sets two phases. The length of the lanes is 60 m. The average velocity of the traffic flow is 2.5 m/s. Let $\alpha = 0.1$, $\gamma = 0.98$, $k = 0.25$, and $T = 40$. Since some general constraints posed by safety rules should be respected when designing the signal plan, we let minimum and maximum green time for each phase be 20 s and 90 s, respectively. The subsets of actions for TSCA 1, TSCA 2, TSCA 3, TSCA 4, and TSCA 5 are given by $A^1 = \{15, 30, 40\}$, $A^2 = \{30, 50\}$, $A^3 = \{35, 45\}$, $A^4 = \{25, 50\}$, and $A^5 = \{30, 35, 40, 55\}$, respectively. Each lane is equally divided into a certain number of segments in an interval of 20 m. As a result, the values of joint state and actions are set as $s = \langle \langle 1, 2, 3, 1 \rangle, \langle 2, 3, 1, 2 \rangle, \langle 2, 1, 2, 1 \rangle, \langle 1, 1, 3, 2 \rangle, \langle 1, 2, 3, 2 \rangle \rangle$, and $a = \langle 30, 30, 45, 25, 35 \rangle$, respectively. Under the state s and in the taken joint action a given above, how the Q -values of TSCA 1, TSCA 2, and TSCA 5 vary with learning time, respectively, was shown in Figure 3.

As can be seen from Figure 3, the multiagent Markov game reinforcement learning approach presented in this paper is convergent; that is, it can reach a Nash equilibrium. In general conditions, an urban traffic network has a relatively stable flow of vehicles, so the time to solve such problem is acceptable.

FIGURE 3: Q-values of TSCA 1, TSCA 2, and TSCA 5 per minute under s and in a .

In the traffic network shown in Figure 2, supposed sometime TSCA 5 is about to choose a red light timing action in north-south direction; then we specify that the local state of TSCA 1, 2, 3, 4 is $s^1 = s^2 = s^3 = s^4 = \langle 1, 1, 1, 1 \rangle$, respectively, so the finally learned Q-values of TSCA 5 are shown in Table 1.

6.2. Analysis of the Method's Effectiveness. We use the average delay time per vehicle ($\bar{d}$) which was defined in (1) of Section 2 as the performance index of each method.

6.2.1. In Comparison with LQF. Local traffic consists of vehicles that cross a single intersection and then exit the network, thereby interacting with just one learning agent. According to [24], when the saturation is greater than 0.90, the level of intersection's service is unbearable. So we think that a highly saturated condition in this paper refers to one in which the saturation greatly exceeds 0.9, and as a result the road is congested and the service level is relatively poor. In this section, we compare the novel approach described in Section 4.2 to LQF traffic signal scheduling algorithm proposed in [25] for an isolated intersection. The LQF algorithm was designed for a signal control problem employing concepts drawn from the field of packet switching in computer networks. It utilized a maximal weight matching algorithm to minimize the queue sizes at each approach yielding significantly lower average vehicle delay through the intersection. The primary limitation of LQF is that every agent only considers its own local traffic volume and thus controls its traffic signals in isolation. Consequently, agents may select individual actions that are locally optimal but that together result in global inefficiencies. Therefore, we focus our experiments on comparisons between the novel method and LQF in highly saturated conditions.

In particular, we also consider the scenery shown in Figure 2. In such traffic network all routes contain at least two intersections, and destinations are selected uniformly, thereby eliminating local traffic. The scenery is challenging

FIGURE 4: The results comparison with LQF.

TABLE 1: The learned Q-values of TSCA 5 in specified state.

Local state of TSCA 5	1, 1, 2, 1	1, 1, 2, 2	1, 1, 2, 3	1, 1, 3, 1
Max Q	231.4	278.9	297.4	211.8
Timing	60	30	25	25

and realistic, as it requires the methods to cope with an abundance of nonlocal traffic. The experiment is designed to test the hypothesis that, under highly saturated conditions, coordinated learning is beneficial when the amount of local traffic is small. If this hypothesis is correct, coordinated learning with multiagent Markov game reinforcement learning should substantially outperform LQF when most vehicles pass through multiple intersections.

We use the average delay time per vehicle ($\bar{d}$) which is defined in (1) of Section 2 as the performance index of each method.

The result is averaged over 10 independent runs. Figure 4 shows the results from the nonuniform destinations and non-local traffic scenery. As can be seen from the figure, multiagent Markov game reinforcement learning substantially outperforms the other noncoordinated method. This result is not surprising since the lack of uniform destinations and local traffic create clear incentive for the TSCAs to learn to coordinate their actions. This approach allows the TSCAs to learn different state transition probabilities and value functions when the outbound lanes are congested. For example, the lane from intersection 1 to 5 is likely to become saturated as all traffic from edge nodes connected to intersection 1 must travel through it. When such saturation occurs, it is important for the two TSCAs to learn to coordinate since allowing incoming traffic to cross intersection 1 is pointless unless intersection 5 allows that same traffic to cross in a "green wave". The cost of including such congestion information is a larger state space and potentially slower learning. So, the simulation results show that multiagent Markov game reinforcement learning is effective.

FIGURE 5: The results comparison with fixed timing control, independent Q reinforcement learning.

6.2.2. In Comparison with Fixed Timing Control Approach and Independent Q Reinforcement Learning Approach. In this section, we compare the novel approach described in Section 4.2 to fixed timing control and independent reinforcement learning. We consider the scenery shown in Figure 2 again. Assume that the vehicle arrival rate in the direction of north-south is

$$p_{sn} = 0.3 + 0.1 \cos\left(\frac{\pi t}{540}\right) \quad (30)$$

and the vehicle arrival rate in the direction of east-west is

$$p_{ew} = 0.1 + 0.5 \sin\left(\frac{\pi t}{540}\right). \quad (31)$$

The yellow light time length is 4 s; TSCA schedules traffic signal every two seconds. The learning time length is 5400 s. During the independent reinforcement learning, TSCA does not consider other TSCA's actions and states, the reinforcement signals from the belief allocation modules are only associated with their own states and actions, then the goal is to maximize the local rewards. In the fixed timing control, the signal cycle length is 120 s. The effective green time length in the direction of north-south and east-west is 56 s. Figure 5 shows the results.

The results show that in low traffic flow the differences between the effectiveness of the three control methods are not particularly evident. When the traffic flow increases gradually, the differences between the performance of the three methods are more and more apparent. Since independent Q-learning process does not take other TSCA's states and actions into consideration, it is not easy to achieve global optimum, which may lower the performance of the system. When the vehicle arrival rate is more than 1000 vehicle/h, independent Q-learning leads to a lot of heavy traffic. On the contrary, in our method, each TSCA must consider the influence of other

TSCA's states and actions, so the results have a certain amount of global properties.

Next, we analyze why multiagent Markov game reinforcement learning approach outperforms the other two approaches. Fixed timing control method can not be adapted to the changes of the traffic environment. In the independent Q-learning algorithm, each TSCA learns and decides at the local level (i.e., using its local state and local action) by using (10). Multiagent Markov game reinforcement learning method biases action selection toward actions that are likely to result in good rewards. The likelihood of good values is evaluated using models of the other agents estimated by the learner through observing their behavior in the past. The efficiency of multiagent Markov game reinforcement learning approach is more profound in cases of traffic fluctuations which assure the adaptability of the approach as the highly saturated condition triggers the TSCAs to coordinate their actions.

7. Conclusion

Previous work about urban traffic control used multiagent reinforcement learning, but the TSCAs selected only locally optimal actions without coordinating their behavior and needed perfect observed information when interacting with environment. In this paper, a multiagent Markov game reinforcement learning approach based on n -player nonzero-sum Markov game is designed for optimizing urban traffic on the basis of the analysis of TSCA's structure model and single-agent Q-learning. Theoretical analysis and experimental result show that the proposed method is convergent and effective. Multiagent Markov game reinforcement learning substantially outperforms the other non-coordinated method like LQE, fixed timing control, and independent reinforcement learning especially under highly saturated conditions when the amount of local traffic is small. It will be demonstrated that the novel method offers the capability to provide distributed control as needed for scheduling multiple intersections.

In future work, it would be interesting to incorporate the effects of driver behavior and transit signal priority in our framework. Moreover, the basic five-intersection network considered here will be expanded to include larger traffic networks and more extensive collaboration among TSCAs.

Acknowledgments

The work described in this paper was supported by the Natural Science Foundation of China (nos. 61263024 and 51268017). The authors would like to thank the members of the academic team guided by Professor Lun-hui Xu for their advice.

References

- [1] C. Diakaki, M. Papageorgiou, and K. Aboudolas, "A multivariable regulator approach to traffic-responsive network-wide signal control," *Control Engineering Practice*, vol. 10, no. 2, pp. 183–195, 2002.

- [2] P. G. Balaji, D. Srinivasan, and C.-K. Tham, "Coordination in distributed multi-agent system using type-2 fuzzy decision systems," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ '08)*, pp. 2291–2298, Piscataway, NJ, USA, June 2008.
- [3] D. Srinivasan, M. C. Choy, and R. L. Cheu, "Neural networks for real-time traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 3, pp. 261–272, 2006.
- [4] J. J. Sánchez, M. Galán, and E. Rubio, "Genetic algorithms and cellular automata: a new architecture for traffic light cycles optimization," in *Proceedings of the Congress on Evolutionary Computation (CEC '04)*, pp. 1668–1674, June 2004.
- [5] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Boston, Mass, USA, 1998.
- [6] M. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Proceedings of the 17th International Conference on Machine Learning and Applications*, pp. 1151–1158, 2000.
- [7] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.
- [8] M. D. Pendrith, "Distributed reinforcement learning for a traffic engineering application," in *Proceedings of the 4th International Conference on Autonomous Agents*, pp. 404–411, ACM Press, Rockville, Md, USA, June 2000.
- [9] B. Bakker, M. Steingrover, R. Schouten, E. Nijhuis, and L. Kester, "Cooperative multi-agent reinforcement learning of traffic lights," in *Proceedings of the Workshop on Cooperative Multi-Agent Learning*, pp. 24–36, 2005.
- [10] A. Salkham, R. Cunningham, A. Garg, and V. Cahill, "A collaborative reinforcement learning approach to urban traffic control optimization," in *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT '08)*, pp. 560–566, Sydney, Australia, December 2008.
- [11] I. Arel, C. Liu, T. Urbanik, and A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," *IET Intelligent Transport Systems*, vol. 4, no. 2, pp. 128–135, 2010.
- [12] J. C. Medina and R. F. Benekohal, "Q-learning and approximate dynamic programming for traffic control—a case study for an oversaturated network," in *Proceedings of the Boar Annual Meeting*, 2012.
- [13] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Traffic light control in non-stationary environments based on multi agent Q-learning," in *Proceedings of the 14th IEEE International Intelligent Transportation Systems Conference (ITSC '11)*, pp. 1580–1585, Washington, DC, USA, October 2011.
- [14] P. G. Balaji, X. German, and D. Srinivasan, "Urban traffic signal control using reinforcement learning agents," *IET Intelligent Transport Systems*, vol. 4, no. 3, pp. 177–188, 2010.
- [15] L. Kuyer, S. Whiteson, B. Bakker, and N. Vlassis, "Multiagent reinforcement learning for urban traffic control using coordination graph," in *Proceedings of the 19th European Conference on Machine Learning*, 2008.
- [16] S. E. Tantawy and B. Abdulhai, "Multi-agent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC)," in *Proceedings of the 15th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC '12)*, pp. 319–326, Anchorage, Alaska, USA, 2012.
- [17] P. Koonce, "Traffic signal timing manual," US Department of Transportation FHWA-HOP-08-024, Federal Highway Administration, 2008.
- [18] M. L. Littman, "Markov games as a framework for multiagent reinforcement learning," in *Proceedings of the 11th International Conference on Machine Learning*, pp. 157–153, Morgan Kaufmann, Boston, Mass, USA, 1994.
- [19] I. Alvarez, A. Poznyak, and A. Malo, "Urban traffic control problem a game theory approach," in *Proceedings of the 47th IEEE Conference on Decision and Control (CDC '08)*, pp. 2168–2172, Cancún, Mexico, December 2008.
- [20] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of Machine Learning Research (JMLR)*, vol. 4, no. 6, pp. 1039–1069, 2003.
- [21] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
- [22] C. Szepesvári and M. L. Littman, "A unified analysis of value-function-based reinforcement-learning algorithms," *Neural Computation*, vol. 11, no. 8, pp. 2017–2060, 1999.
- [23] J. Filar and K. Vrieze, *Competitive Markov Decision Processes*, Springer, New York, NY, USA, 1997.
- [24] W. Wang, *Urban Traffic Planning*, Southeast University Press, Nanjing, China, 1999.
- [25] R. Wunderlich, C. Liu, I. Elhanany, and T. Urbanik II, "A novel signal-scheduling algorithm with quality-of-service provisioning for an isolated intersection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 536–547, 2008.