Cooperative Traffic Signal Online Control Using Game Theory and Contextual Bandit

Junjie Shao, Yu Zhao, He Sun, Jinbo Cai, Jie Wu, *Fellow, IEEE,* and Mingjun Xiao, *Member, IEEE*

Abstract—Due to the surge in global traffic volume, traffic congestion has become a critical issue. In response, researchers have been actively searching for a more effective traffic signal control service. In recent years, several methods based on Deep Reinforcement Learning (DRL) have been proposed to tackle the traffic signal control problem. While these methods have demonstrated good performance in simulation environments, practical implementation still faces some challenges. Firstly, DRL-based methods often require a significant amount of data and computational resources and a large number of iterations to yield impressive results. Secondly, most DRL-based methods neglect the fairness factor, leading to imbalanced waiting times for vehicles. Unlike these state-of-the-art DRL-based approaches, we take the cooperation between intersections as well as fairness into consideration and propose a novel traffic signal online control algorithm, called FCTSC. By exploiting the contextual MAB model combined with game theory, FCTSC can achieve a fair and efficient solution quickly. Moreover, we provide a comprehensive theoretical analysis of our proposed algorithm and demonstrate its superiority through extensive simulations.

Index Terms—Intelligent traffic service, Game theory, Contextual multi-armed bandit, Multi-agent system, Traffic signal control

1 Introduction

With the rapid urbanization, the increase in traffic volume in large cities has put a strain on the road system, leading to increasingly severe traffic congestion. Traffic congestion is a significant social problem that has a profound impact on urban development. According to statistics from [1], the average commuter in Dublin spent over 277 hours traveling in 2022, equivalent to more than an hour per working day lost to travel, with approximately half of that time attributed to traffic congestion. Traffic congestion not only exacerbates overcrowding but also results in substantial economic losses. As shown in [2], traffic congestion costs the U.S. economy over 120 billion dollars annually. Moreover, traffic congestion contributes to unnecessary fuel consumption by vehicles, exacerbating environmental pollution.

To reduce the impact of traffic congestion, it is necessary to optimize urban transportation systems. Existing methods mainly include expanding and optimizing the road network, promoting public transportation, and developing intelligent transportation systems. In the construction of intelligent transportation systems, traffic conditions near intersections largely determine the overall traffic situation of the entire city, greatly influencing the congestion on roads. Therefore, deploying traffic signal control at intersections plays a vital role in urban traffic management services.

Over the past few decades, researchers have proposed various traffic signal control methods. Although some early methods, like SCATS and SCOOT [3], have been widely applied, they heavily rely on a given traffic model or depend on pre-defined rules based on expert knowledge. In recent years, Deep Reinforcement Learning (DRL) methods have emerged as a popular approach to tackle the challenges of traffic signal control. These DRL-based methods offer the capability of learning optimal control policies through interaction with the environment, such as IDQN [4] and PressLight [5], having demonstrated significant performances.

However, despite their advantages, DRL-based methods for traffic signal control still have some limitations. These methods generally require a large amount of data and long-term training, which will continuously consume huge computation and storage resources, significantly exceeding the capabilities of typical roadside infrastructures. Consequently, they can only provide the traffic signal control models through offline training, which are generally unable to keep pace with changing traffic flows, lacking the adaptability to real-time dynamic application scenarios.

To offer an online traffic signal control service, we focus on how to employ the contextual Multi-Armed Bandit (MAB) model to train traffic signal control strategies. MAB finds extensive application in various domains such as crowdsensing [6]–[10] and auction theory [11], [12]. Compared to traditional DRL-based approaches, the contextual MAB model can converge to a desired traffic signal control strategy quickly without conducting the offline training on a large amount of data in advance. Within the contextual MAB framework, we take two challenges into consideration. Firstly, the traffic signal control strategy needs to ensure fairness, i.e., distributing waiting time equitably among different lanes to prevent excessive delays in some specific

- *J. Shao, Y. Zhao, H. Sun, J. Cai, and M. Xiao (Corresponding author) are with the School of Computer Science and Technology / Suzhou Institute for Advanced Research / State Key Laboratory of Cognitive Intelligence, University of Science and Technology of China (USTC), Hefei, China. E-mail: {fdssjj@mail., zhaoyu0624@mail., hesun@mail., SA21011121@mail., xiaomj@}ustc.edu.cn.*
- *J. Wu is with the Center for Networked Computing, Temple University, Philadelphia, PA 19122, USA. E-mail: jiewu@temple.edu.*

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants 62172386, 61872330, 61572457, the Natural Science Foundation of Jiangsu Province in China under Grants BK20231212, BK20191194, BK20131174, and the Teaching Research Project of the Education Department of Anhui Province in China (No. 2021jyxm1738).

direction or road user. Although fairness has been extensively explored in various domains such as deep learning (e.g., [13]–[15]) and crowdsensing (e.g., [16]–[18]), its consideration in the context of traffic signal control has been notably limited. Secondly, the traffic signal control strategy also needs to enable effective cooperation between neighboring intersections, so as to optimize traffic flow and reduce congestion on the entire road network.

To tackle the above challenges, we model the fair cooperative traffic signal control problem as a contextual MAB issue combined with an incomplete information multi-agent game as well as a fairness constraint. Then, the techniques of ridge regression, the Upper Confidence Bound (UCB) index, virtual queue, and fictitious play are integrated to deal with the fair contextual MAB issue with an incomplete information game, based on which we propose a Fair Cooperative Traffic Signal Control approach, called FCTSC. The detailed contributions of the paper are summarized as follows:

- We introduce an online traffic signal control problem which takes fairness and the cooperation between neighboring intersections into consideration. Moreover, we model this problem as a contextual MAB issue together with an incomplete information multi-agent game and a fairness constraint to be solved.
- We propose the FCTSC algorithm where the techniques of the ridge regression, the Upper Confidence Bound (UCB) index, and virtual queue are combined to address the fair contextual bandit issue, while fictitious play is employed to solve the incomplete information game issue. Consequently, FCTSC can ensure fairness as well as enable cooperation between intersections.
- We provide a comprehensive theoretical analysis of the proposed FCTSC algorithm, including the proof of fairness, the convergence to the equilibrium, and the regret.
- We conduct extensive simulations on both synthetic and real-world datasets to demonstrate that the proposed FCTSC algorithm outperforms several state-of-the-art methods.

2 Related Works

Traffic signal control presents a significant challenge in the field of transportation. A comprehensive overview of traditional algorithms for traffic signal control is provided in the survey by Wei et al. [3], empassing SCATS, SCOOT, GreenWave and MaxPressure. SCATS and SCOOT optimize traffic flow by adaptively adjusting the control plans of traffic signals. GreenWave optimizes traffic flow in a single direction on a main road by adjusting the offsets between intersections. MaxPressure is considered a state-of-the-art signal control method that maximizes the throughput of the network by greedily selecting actions. In recent years, machine learning techniques have gained prominence in tackling the traffic signal control problem. Various approaches, including fuzzy logic algorithms [19], swarm intelligence [20], and reinforcement learning [4], [5], [21]–[27], have been applied. Among these approaches, reinforcement learning has emerged as the most popular approach for traffic signal control. However, it is worth noting that the mainstream approaches using deep reinforcement learning do not account for the fairness of waiting times and often fail to meet real-world application requirements in terms of convergence speed and other performance metrics.

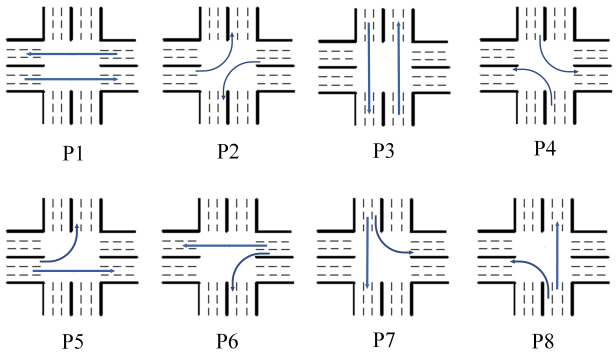

Fig. 1: The illustration of eight phases in a simple four-way intersection.

Cooperation between intersections is a crucial aspect of traffic signal control when employing reinforcement learning techniques. In RL-based methods, a common approach to achieving cooperation is to employ a global central agent responsible for controlling the traffic signal schemes of all intersections [28]. However, this method faces the curse of dimensionality, as the size of the state and action space exponentially increases with the number of intersections in the system. To mitigate this challenge, most existing research on RL-based traffic signal control adopts separate agents for each intersection. Some studies, such as [24] and [25], enable agents to not only observe their own intersection but also communicate with other intersections to obtain information, such as traffic flow, thereby facilitating cooperation. Building upon this, the Colight model [26] considers the strength of influence between intersections and employs a graph attention network to learn these differences, leading to improved cooperation. However, none of these works have considered combining game theory with reinforcement learning.

Unlike these works, we focus on the online traffic signal control issue in this paper, where the fairness and the cooperation between intersections, which is seen as an incomplete information multi-agent game, are taken into consideration simultaneously.

3 System Overview and Problem

3.1 System Overview

In this paper, we investigate a traffic signal control system that consists of N intersections. We refer to the edge nodes responsible for controlling the traffic signals at each intersection as agents. The system runs for a duration of T rounds, where each round lasts Δt seconds. To define the agents' actions in each round, we define two concepts: "traffic movement signal" and "phase".

Definition 1 (Traffic Movement Signal and Phase)**.** In a traffic signal control system, a traffic movement signal regulates a specific path for vehicles to cross an intersection,

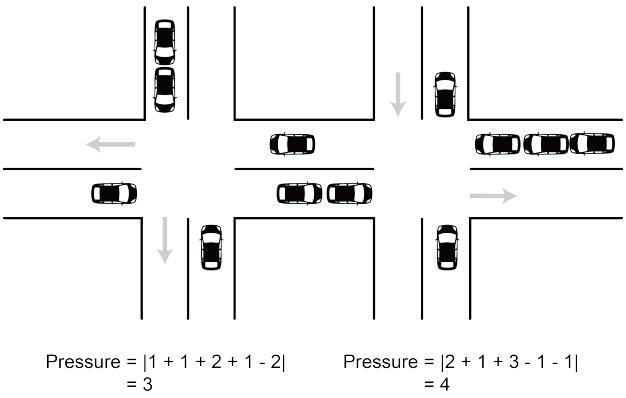

Fig. 2: Illustration of intersection pressure. For clarity, only straight driving is allowed in this example.

typically from an entering lane to an exiting lane. A phase represents a set of traffic movement signals that can be executed simultaneously without conflicts.

Fig. 1 illustrates a set of eight phases commonly used in a typical four-way intersection to cover all movement signals. It is worth noting that right-turn movement signals generally do not conflict with other movements and are included in every phase by default.

We use S and $\mathcal{A}=(a_1, a_2, \ldots, a_k, \ldots, a_{|A|})$ to denote the set of traffic movement signals and the set of phases, respectively. For simplicity, we assume that all intersections share the same sets of S and $\mathcal{A}$ in this paper. It is worth noting that our model and algorithms can be easily extended to the scenario where different intersections have heterogeneous sets S and $\mathcal{A}$. In each round t, agent i observes the surrounding environment of the intersection through the sensors and cameras installed at the intersection, so as to obtain an observation, denoted as $\boldsymbol{o}_{t,i} = (o_{t,i,1}, o_{t,i,2}, \ldots, o_{t,i,d})^\top$, where d is the dimensionality of the observation and $\top$ means the transposition operation. Here, we let each component of $\boldsymbol{o}_{t,i}$ record the number of vehicles in a lane around intersection i. In practice, it can also include other additional information such as the average vehicle speed. Based on this observation, agent i will select a phase from $\mathcal{A}$ as its action of this round, denoted as $a_{t,i}$, which is a set of several traffic movement signals. Consequently, the selection will produce a reward which indicates the traffic congestion level in intersection i, denoted as $r_{t,i}$.

In this paper, like many existing works (e.g., [5]), we use the concept of "pressure" to indicate the reward rather than the average travel time. Here, the pressure of an intersection refers to the difference between the numbers of incoming and outgoing vehicles at that intersection. Fig. 2 provides an illustration of intersection pressure, where the left intersection has a pressure of 3, and the right intersection has a pressure of 4. According to the pressure, the agent can make the traffic signal control decisions based on some strategy to alleviate congestion by evenly distributing traffic across intersections. Specifically, the reward function of the traffic signal control is defined as follows:

$$r_{t,i} = -P_{t,i}, \tag{1}$$

TABLE 1: DESCRIPTION OF MAJOR NOTATIONS

Symbol	Description
N	the number of intersections/agents.
$\mathcal{N}_i$	the set of intersection i's neighboring intersections.
M	the maximum number of neighboring intersections.
$\Delta t, T$	the duration of a round and the total number of rounds
$\boldsymbol{o}_{t,i}$	the observation of agent i in round t
$a_{t,i}, \mathcal{A}$	the chosen phase of intersection i in round t and the available phase set
S	the set of feasible movement signals
$g_{i,s}(t)$	a function which indicates whether movement signal s is selected in round t for intersection i, where $g_{i,s}(t) = 1$ indicates that s is selected and $g_{i,s}(t) = 0$ indicates that s is not selected
$P_{t,i}$	the pressure of intersection i at round t
$r_{t,i}$	the reward for agent i in round t
$\boldsymbol{\phi}$	a bandit policy, which indicates a sequence of chosen phases/arms
$f_{i,s}$	required minimum selection fraction for movement signal s
$\mathcal{C}$	maximal feasibility region

where $P_{t,i}$ represents the pressure of intersection i at time t. In this paper, we assume that the reward $r_{t,i}$ falls within the range $[r_{min}, r_{max}]$.

In addition, fairness is also taken into consideration. Specifically, we need to ensure that each movement signal is selected in a certain fraction of rounds on average, no less than a predefined threshold. We use $g_{i,s}(t)$ to indicate whether movement signal s is selected in round t by agent i. Let $g_{i,s}(t) = 1$ if movement signal s is selected, and $g_{i,s}(t) = 0$ otherwise. Then, the selection of movement signals should satisfy

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[g_{i,s}(t)] \geq f_{i,s} \quad \forall s \in S, i = 1, 2, \ldots, N, \tag{2}$$

where $f_{i,s} \in (0, 1)$ represents the required minimum fraction of rounds in which movement signal s is selected for intersection i to ensure fairness. A fair algorithm should strive to satisfy Eq. (2) for as many predefined minimum fraction thresholds as possible. This motivates the following definition:

Definition 2 (Fairness)**.** For agent i, the minimum selection fraction vector $\boldsymbol{f}_i = (f_{i,1}, \ldots, f_{i,|S|})$ is considered feasible if there exists an algorithm that can satisfy Eq. (2). The maximal feasibility region $\mathcal{C}$ is the set of all such feasible vectors. An algorithm is considered to be fair if it can satisfy any vector $\boldsymbol{f}_i$ that lies strictly inside the maximal feasibility region $\mathcal{C}$.

According to Def. 2, a fair algorithm can guarantee that each lane has a fair chance of being selected, preventing long waiting times for the lanes with low arrival rates. This enhances the overall efficiency and equity of the traffic flow.

3.2 Contextual MAB Modeling and Problem Formulation

During each round, each agent is confronted with selecting the most suitable phase. We model it as a Contextual MAB problem to be solved, where each phase in $\mathcal{A}$ is considered as an "arm" of the bandit, and pulling an arm is equivalent to choosing the corresponding phase. The reward of pulling an arm is defined in Eq. (1). Here, the context information refers to the observations $\boldsymbol{o}_{t,i}$ made by each agent. Our goal is to determine a bandit policy, whereby each agent can explore and exploit the arm to yield the highest accumulated reward based on its context information.

The bandit policy of agent i is defined as $\boldsymbol{\phi}_i = (a_{1,i}, a_{2,i}, \ldots, a_{T,i})$, where $a_{t,i}$ is the chosen phase of policy $\boldsymbol{\phi}_i$ in round $t \in [1, T]$. The accumulated reward of $\boldsymbol{\phi}_i$ is

$$R(\boldsymbol{\phi}_i) = \sum_{t=1}^{T} r_{t,i}. \tag{3}$$

We use $\mu_i(a_k|\boldsymbol{o}_{t,i})$ to denote the expected reward of phase (i.e., arm) a_k being selected when given an observation $\boldsymbol{o}_{t,i}$. Here, like in [29], we assume that for each agent i, $\mu_i(a_k|\boldsymbol{o}_{t,i})$ is a linear combination of the observation $\boldsymbol{o}_{t,i}$ with an unknown coefficient vector $\boldsymbol{\theta}_{i,a_k}$ as follows:

$$\mu_i(a_k|\boldsymbol{o}_{t,i}) = \boldsymbol{o}_{t,i}^{\top}\boldsymbol{\theta}_{i,a_k}, \forall a_k \in \mathcal{A} \tag{4}$$

Each agent i attempts to find the optimal policy $\boldsymbol{\phi}_i$ to maximize the expected accumulated reward while satisfying the fairness constraints defined in Eq. (2), which is formulated as follows:

$$\text{Maximize:} \quad \mathbb{E}[R(\boldsymbol{\phi}_i)] = \sum_{t=1}^{T} \mu_i(a_{t,i}|\boldsymbol{o}_{t,i}) \tag{5}$$

$$\text{Subject to:} \quad \frac{1}{T}\sum_{t=1}^{T} g_{i,s}(t) \geq f_{i,s}, \quad \forall s \in S \tag{6}$$

$$a_{t,i} \in \mathcal{A}, \quad \forall t, \tag{7}$$

Since the action of each intersection might influence the pressure of neighboring intersections, the bandit policies of neighboring agents will influence each other. To address the bandit issue of multiple agents, we extend the above model into the multiple intersections scenario by treating it as a multi-agent perturbed game with incomplete information. In this game, each agent does not know the current phase choices of neighboring agents. We denote the probability distribution of the joint phases of agent i's neighboring agents in round t as $\mathbb{P}_{t,i}$. Now, the expected reward of each phase is also correlated with $\mathbb{P}_{t,i}$, so we can rewrite it as $\mu_i(a_{t,i}|\boldsymbol{o}_{t,i}, \mathbb{P}_{t,i})$. Then, in this incomplete information multi-agent game, the problem of finding the optimal bandit policy for each agent i can be reformulated as follows:

$$\text{Maximize:} \quad \sum_{t=1}^{T} \mu_i(a_{t,i}|\boldsymbol{o}_{t,i}, \mathbb{P}_{t,i}) \tag{8}$$

$$\text{Subject to:} \quad \frac{1}{T}\sum_{t=1}^{T} g_{i,s}(t) \geq f_{i,s}, \quad \forall s \in S \tag{9}$$

$$a_{t,i} \in \mathcal{A}, \quad \forall t. \tag{10}$$

Additionally, we say that this game converges to an equilibrium if it satisfies:

$$\lim_{t \longrightarrow \inf} \max\left(\mu_i(a_{t,i}|\boldsymbol{o}_{t,i}, \mathbb{P}_{t,i}) - \mu_i(a_{t,i}^{*}|\boldsymbol{o}_{t,i}, \mathbb{P}_{t,i})\right) = 0, \quad i = 1, 2, \ldots, N. \tag{11}$$

Here, $a_{t,i}^{*}$ is the optimal bandit strategy. Eq. (11) means that if agent i chooses any phase $a_{t,i}$ rather than the phase $a_{t,i}^{*}$, it will achieve a lower reward, so that no agent is willing to select other phases, forming an equilibrium.

4 The FCTSC Algorithm

In this section, we present the FCTSC algorithm to address the traffic signal control problem, grounded in the contextual MAB framework within an incomplete information multi-agent game. The basic idea is as follows: Firstly, inspired by the LinUCB algorithm [30], we leverage ridge regression and the Upper Confidence Bound (UCB) index to address the above contextual MAB problem. Secondly, considering that our contextual MAB model contains the constraint of fairness unlike LinUCB, we employ virtual queue techniques to deal with the fairness issue. Thirdly, we address the complexities that arise from the incomplete information multi-agent game by utilizing the fictitious play strategy as a key mechanism to foster the cooperation between intersections. The concrete solution and the detailed algorithm are presented as follows.

Firstly, we use the techniques of ridge regression and the UCB index to estimate rewards while carefully balancing the exploration and exploitation for our traffic signal control strategy.

We begin with predicting the reward of every phase (i.e., arm) $a_k \in \mathcal{A}$ in intersection i at round t. Suppose that a_k has been selected m times, resulting in a collection of m observations and rewards. With these observations and rewards, we can use ridge regression to estimate the corresponding coefficient vector $\boldsymbol{\theta}_{i,a_k}$ in Eq. (4). We systematically arrange these context vectors as rows to create a matrix denoted as $\mathbf{D}_{a_k}$ with dimension $m \times d$, where d is the dimensionality of the observation. Likewise, the m rewards are consolidated into a vector of size m designated as $\mathbf{c}_{a_k}$. Let $\hat{\boldsymbol{\theta}}_{i,a_k}$ denote the estimation of $\boldsymbol{\theta}_{i,a_k}$. Then, it can be achieved through the following equation:

$$\hat{\boldsymbol{\theta}}_{i,a_k} = (\mathbf{D}_{a_k}^{\top}\mathbf{D}_{a_k} + \mathbf{I}_d)^{-1}\mathbf{D}_{a_k}^{\top}\mathbf{c}_{a_k}, \tag{12}$$

where $\mathbf{I}_d$ denotes an identity matrix of size d.

Using the estimated coefficient vector $\hat{\boldsymbol{\theta}}_{i,a_k}$, we obtain the estimated reward $\hat{\mu}_i(a_k|\boldsymbol{o}_{t,i}) = \boldsymbol{o}_{t,i}^{\top}\hat{\boldsymbol{\theta}}_{i,a_k}$ and calculate the UCB value for pulling the arm $a_k \in \mathcal{A}$, i.e., $a_{t,i} = a_k$, with the following formula [30]:

$$UCB_{t,i}(a_k|\boldsymbol{o}_{t,i}) = min\{max\{\hat{\mu}_i(a_k|\boldsymbol{o}_{t,i}) + \alpha\sqrt{\boldsymbol{o}_{t,i}^{\top}(\boldsymbol{A}_{a_k})^{-1}\boldsymbol{o}_{t,i}}, r_{min}\}, r_{max}\}, \tag{13}$$

where $\boldsymbol{A}_{a_k} = \boldsymbol{D}_{a_k}^{\top}\boldsymbol{D}_{a_k} + I_d$, $\alpha = 1 + \sqrt{\ln(2/\delta)/2}$, and $\delta > 0$ is a parameter that indicates the confidence level. $\mu_i(a_k|\boldsymbol{o}_{t,i}) \leq UCB_{t,i}(a_k|\boldsymbol{o}_{t,i})$ holds with a high probability of at least $1 - \delta$. By adjusting the value of α, we can tailor the algorithm's behavior to effectively address the trade-off

between exploring new phases to gather information and exploiting the known phases to maximize rewards.

Secondly, after obtaining the UCB values, we introduce the concept of virtual queues to enforce fairness constraints among movement signals at each intersection. We define $Q_{i,s}(t)$ to represent the length of the virtual queue associated with movement signal s for agent i in round t. The evolution of the virtual queue length $Q_{i,s}(t)$ is determined by the formula:

$$Q_{i,s}(t) = [Q_{i,s}(t-1) + f_{i,s} - g_{i,s}(t-1)]^+, \quad (14)$$

where $[x]^+ \stackrel{\text{def}}{=} max\{x, 0\}$. At the beginning, the virtual queue lengths for all agents i and movement signals s are initialized to 0, i.e., $Q_{i,s}(t=1) = 0$. In each subsequent round, the virtual queue length increases by $f_{i,s}$, which represents the minimum selection fraction of movement signal s at intersection i. Additionally, the virtual queue length decreases by one if movement signal s was selected in the previous round $t-1$, i.e., $g_{i,s}(t-1) = 1$. These virtual queue lengths represent the extent to which the corresponding movement signals meet the minimum requirements. A larger virtual queue length for a movement signal indicates that it has not been selected frequently enough in the past, and thus, it needs a higher priority compared to the movement signals with smaller virtual queue lengths.

By combining the UCB value and virtual queue lengths, we can define the fairness-sensitive UCB index, called FUCB, as follows:

$$FUCB_{t,i}(a_k|\boldsymbol{o}_{t,i}) = \eta UCB_{t,i}(a_k|\boldsymbol{o}_{t,i}) + \sum_{s\in a_k} Q_{i,s}(t). \quad (15)$$

Here, η controls the balance between the reward and the virtual queue lengths, providing the flexibility of adjusting the trade-off between the reward and the fairness, which enables the adaptability to different traffic scenarios. By utilizing FUCB, agents can effectively make decisions that promote fairness while optimizing the accumulated reward.

Thirdly, we employ a classic approach known as the fictitious play to address the incomplete information multi-agent game problem, in which intersections cannot access the decisions made by neighboring intersections in advance. More specifically, for each intersection i and its neighboring intersections, we begin with constructing the reward matrix $\boldsymbol{M}_{t,i}$. Let $\mathcal{N}_i$ denote the neighboring intersections of intersection i and let $\mathcal{A}_{\mathcal{N}_i} = \mathcal{A}^{|\mathcal{N}_i|}$ denote the set of available joint phases of $\mathcal{N}_i$. Let $\boldsymbol{o}'_{t,i,h} = (o'_{t,i,h,1}, o'_{t,i,h,2}, \ldots, o'_{t,i,h,d'})^\top$ denote the additional observation which contains the expected number of vehicles in each lane that will approach intersection i if the neighboring intersections opt for the joint phase $\boldsymbol{a}_h \in \mathcal{A}_{\mathcal{N}_i}$. Then, the corresponding item in the reward matrix $\boldsymbol{M}_{t,i}$ for any $a_k \in \mathcal{A}$ can be calculated as follows:

$$\boldsymbol{M}_{t,i}[k][h] = \begin{bmatrix} \boldsymbol{o}_{t,i} \\ \boldsymbol{o}'_{t,i,h} \end{bmatrix}^\top \begin{bmatrix} \hat{\boldsymbol{\theta}}_{a_k} \\ \hat{\boldsymbol{\theta}}'_{a_k} \end{bmatrix}. \quad (16)$$

Here, $\boldsymbol{M}_{t,i}[k][h]$ represents the estimated reward that agent i can obtain, if agent i chooses phase a_k and its neighboring intersections choose $\boldsymbol{a}_h$. Moreover, $\hat{\boldsymbol{\theta}}'_{a_k}$ is a coefficient vector similar to $\hat{\boldsymbol{\theta}}_{a_k}$ but corresponding to $\boldsymbol{o}'_{t,i,h}$.

Algorithm 1: The FCTSC algorithm.

Input: $\alpha \in \mathbb{R}_+$

for agent i:
// Initialization
foreach $s \in S$ **do**
  Initialize $Q_{i,s}(t) = 0$;
end
foreach $a_k \in \mathcal{A}$ **do**
  $\boldsymbol{A}_{a_k} \leftarrow \boldsymbol{I}_{d+d'}$;
  $\boldsymbol{b}_{a_k} \leftarrow \boldsymbol{0}_{(d+d')\times 1}$;
end
// End initialization
for $t = 1, 2, 3, \ldots, T$ **do**
  Observe feature vector $\boldsymbol{o}_{t,i} \in \mathbb{R}^d$;
  Update $Q_{i,s}(t)$ for all $s \in S$ according to Eq. (14);
  Observe actions of neighbor agents in the last round and update $\hat{\mathbb{P}}_{t,i}$;
  $\bar{\boldsymbol{o}}_{t,i} \leftarrow \begin{bmatrix} \boldsymbol{o}_{t,i} \\ \mathcal{O}'_{t,i}\hat{\mathbb{P}}_{t,i} \end{bmatrix}$
  foreach $a_k \in \mathcal{A}$ **do**
    $\begin{bmatrix} \hat{\boldsymbol{\theta}}_{a_k} \\ \hat{\boldsymbol{\theta}}'_{a_k} \end{bmatrix} \leftarrow \boldsymbol{A}_{a_k}^{-1}\boldsymbol{b}_{a_k}$;
    calculate $FUCB_{t,i}(a_k|\bar{\boldsymbol{o}}_{t,i})$ according to Eq. (18);
  end
  Choose phase $a^*_{t,i}$ according to Eq. (19) with ties broken arbitrarily;
  Observe a real-valued reward $r_{t,i}$;
  $\boldsymbol{A}_{a^*_{t,i}} \leftarrow \boldsymbol{A}_{a^*_{t,i}} + \hat{\boldsymbol{o}}_{t,i}\hat{\boldsymbol{o}}_{t,i}^\top$;
  $\boldsymbol{b}_{a^*_{t,i}} \leftarrow \boldsymbol{b}_{a^*_{t,i}} + r_{t,i}\hat{\boldsymbol{o}}_{t,i}$;
end

After obtaining the reward matrix, each agent i can estimate the probability of each $\boldsymbol{a}_h \in \mathcal{A}_{\mathcal{N}_i}$ to get the probability distribution $\hat{\mathbb{P}}_{t,i}$ by using the fictitious play strategy based on its neighboring agents' historical phases. Then, the estimated expected reward of choosing phase a_k is:

$$\begin{aligned} \hat{\mu}_{t,i}(a_k|\boldsymbol{o}_{t,i}, \hat{\mathbb{P}}_{t,i}) &= \boldsymbol{M}_{t,i}[k]\hat{\mathbb{P}}_{t,i} \\ &= \begin{bmatrix} \boldsymbol{o}_{t,i} \\ \mathcal{O}'_{t,i}\hat{\mathbb{P}}_{t,i} \end{bmatrix}^\top \begin{bmatrix} \hat{\boldsymbol{\theta}}_{a_k} \\ \hat{\boldsymbol{\theta}}'_{a_k} \end{bmatrix} \\ &= \bar{\boldsymbol{o}}_{t,i}^\top \begin{bmatrix} \hat{\boldsymbol{\theta}}_{a_k} \\ \hat{\boldsymbol{\theta}}'_{a_k} \end{bmatrix} \\ &= \hat{\mu}_i(a_k|\bar{\boldsymbol{o}}_{t,i}), \end{aligned} \quad (17)$$

where $\boldsymbol{M}_{t,i}[k]$ represents the k-th row vector in the reward matrix $\boldsymbol{M}_{t,i}$ that phase a_k corresponds to, $\mathcal{O}'_{t,i} = (\boldsymbol{o}'_{t,i,1}, \boldsymbol{o}'_{t,i,2}, \ldots, \boldsymbol{o}'_{t,i,|\mathcal{A}_{\mathcal{N}_i}|})$ aggregates all possible $\boldsymbol{o}'_{t,i,h}$ for h in the range from 1 to $|\mathcal{A}_{\mathcal{N}_i}|$, and $\bar{\boldsymbol{o}}_{t,i}$ denotes the expected observation of agent i in round t. Therefore, instead of computing the entire reward matrix, we can directly calculate the expected observation to simplify the computation. When considering the game between intersections, the FUCB defined in Eq. (15) can be transformed into:

$$FUCB_{t,i}(a_k|\bar{\boldsymbol{o}}_{t,i}) = \eta UCB_{t,i}(a_k|\bar{\boldsymbol{o}}_{t,i}) + \sum_{s\in a_k} Q_{i,s}(t). \quad (18)$$

Now, we can select the phase using the following equation:

$$a_{t,i}^{*} = \underset{a_k \in \mathcal{A}}{\operatorname{argmax}} \left(FUCB_{t,i}(a_k)\right). \tag{19}$$

Following the above basic idea, we present the detailed FCTSC algorithm, as shown in Algorithm 1. At the beginning, all virtual queues $Q_{i,s}(t)$ for each movement signal s of agent i are initialized to 0 (Steps 3-5). The matrices $\boldsymbol{A}_{a_k}$ and vectors $\boldsymbol{b}_{a_k}$ are also initialized for ridge regression where $\boldsymbol{b}_{a_k} = \boldsymbol{D}_{a_k}^{\top}\boldsymbol{c}_{a_k}$ (Steps 6-9). In each round t, agent i gets its observation $\boldsymbol{o}_{t,i}$ and updates the virtual queue lengths $Q_{i,s}(t)$ for each movement signal $s \in S$ (Steps 12-13). Subsequently, agent i collects the information about its neighbors' phases from the previous round and updates the probability distributions of actions for all neighboring agents $\hat{\mathbb{P}}_{t,i}$. Next, agent i chooses the optimal phase $a_{t,i}^{*}$ according to Eq. (19) (Step 16). Then, it receives a reward (Step 17) and updates $\boldsymbol{A}_{a_{t,i}^{*}}$ and $\boldsymbol{b}_{a_{t,i}^{*}}$ (Steps 18-20). The algorithm continues in this manner for subsequent rounds. By iteratively updating the virtual queue lengths, estimating the probability distributions of neighboring agents' actions, and selecting actions based on both the expected reward and virtual queue lengths, the FCTSC algorithm effectively achieves fairness and cooperation between intersections in a decentralized manner.

The computation complexity of FCTSC is $O(TN(d+d')^2|\mathcal{A}|^M)$, where T is the number of rounds, N is the number of intersections, d and d' are the sizes of observations $\boldsymbol{o}_{t,i}$ and $\boldsymbol{o}'_{t,i,h}$, and M is the maximum number of neighboring intersections. In practical scenarios, the number of neighboring intersections for a given intersection is generally a very small integer, implying that $N(d+d')^2|\mathcal{A}|^M$ actually is a limited constant. Therefore, the overall time complexity of FCTSC is linear with the number of rounds T. This computation complexity ensures that the FCTSC algorithm is scalable and computationally feasible, making it suitable for large-scale traffic signal control problems.

5 Theoretical Analysis

In this section, we present the theoretical analysis for FCTSC. We first prove the fairness and convergence properties of the FCTSC algorithm, and then we conduct an analysis of its regret.

5.1 Fairness Analysis

Firstly, we define a special class of policies called Ω-only policies, which are designed to ensure the fairness constraint. An Ω-only policy for intersection i, denoted as $\boldsymbol{\phi}_i^{\Omega}$, will select each phase from $\mathcal{A}$ according to a fixed probability distribution, which can be expressed as follows:

$$\begin{aligned} P(a_{t,i}^{\Omega} = a_k) &= p_{i,a_k}^{\Omega}, \\ \sum_{a_k \in \mathcal{A}} p_{i,a_k}^{\Omega} &= 1, \end{aligned} \tag{20}$$

where $a_{t,i}^{\Omega}$ denotes the phase that policy $\boldsymbol{\phi}_i^{\Omega}$ selects in round t and p_{i,a_k}^{Ω} represents the probability that policy $\boldsymbol{\phi}_i^{\Omega}$ selects phase a_k. Under policy $\boldsymbol{\phi}_i^{\Omega}$, the expected selection rate for a movement signal s is given by:

$$\mathbb{E}\left[g_{i,s}^{\Omega}(t)\right] = \sum_{s \in a_k} p_{i,a_k}^{\Omega}. \tag{21}$$

Furthermore, we can establish the following lemma:

Lemma 1. For any vector $\boldsymbol{f}_i$ that is strictly inside the maximal feasibility region $\mathcal{C}$, there must exist an Ω-only policy that can support $\boldsymbol{f}_i$.

Proof. According to Definition 2, there exists an algorithm that can satisfy $\boldsymbol{f}_i$. Let us assume that for each phase $a_k \in \mathcal{A}$, this algorithm selects a_k an expected number of n_i times within T rounds. We can construct an Ω-only policy with $p_{i,a_k}^{\Omega} = n_i/T$. Clearly, this policy satisfies $\boldsymbol{f}_i$. □

Secondly, the fairness constraint Eq. (2) is satisfied as long as all virtual queues are mean rate stable, i.e., $\lim_{T\to\inf} \frac{1}{T}\mathbb{E}[|Q_{i,s}(T)|] = 0$ for $\forall s \in S$ and $i = 1, 2, \ldots, N$. So, to prove the fairness of FCTSC, we only need to prove that all virtual queues are mean rate stable. We state the result in the following theorem:

Theorem 1. The FCTSC algorithm is fair. More specifically, for any minimum selection fraction $\boldsymbol{f}_i$ strictly inside the maximal feasibility region $\mathcal{C}$, FCTSC ensures that all virtual queues are mean rate stable.

Proof. For intersection i, we define its queue length vector at round t as $\boldsymbol{Q}_i(t) = (Q_{i,1}(t), \ldots, Q_{i,|S|}(t))$. We then define the following function:

$$L(\boldsymbol{Q}_i(t)) \overset{\text{def}}{=} \frac{1}{2}\sum_{s \in S} Q_{i,s}(t), \tag{22}$$

which is called the Lyapunov function [31]. The one-slot conditional Lyapunov drift $\Delta(L(\boldsymbol{Q}_i(t)))$ is defined as follows:

$$\Delta(L(\boldsymbol{Q}_i(t))) \overset{\text{def}}{=} \mathbb{E}[L(\boldsymbol{Q}_i(t+1)) - L(\boldsymbol{Q}_i(t))|\boldsymbol{Q}_i(t)]. \tag{23}$$

Substituting Eq. (14) into this equation yields:

$$\begin{aligned} &\Delta(L(\boldsymbol{Q}_i(t))) \\ &= \mathbb{E}[L(\boldsymbol{Q}_i(t+1)) - L(\boldsymbol{Q}_i(t))|\boldsymbol{Q}_i(t)] \\ &\le \mathbb{E}\left[\frac{1}{2}\sum_{s\in S}(Q_{i,s}(t) + f_{i,s} - g_{i,s}(t))^2 - \frac{1}{2}\sum_{s\in S}Q_{i,s}^2(t)|\boldsymbol{Q}_i(t)\right] \\ &= \mathbb{E}\left[\frac{1}{2}\sum_{s\in S}(f_{i,s} - g_{i,s}(t))^2 + \sum_{s\in S}(f_{i,s} - g_{i,s}(t))Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right] \\ &\le \mathbb{E}\left[\frac{1}{2}\sum_{s\in S}1 + \sum_{s\in S}(f_{i,s} - g_{i,s}(t))Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right] \\ &= \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) - \mathbb{E}\left[\sum_{s\in S} g_{i,s}(t)Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right]. \end{aligned} \tag{24}$$

To simplify the notation, we use $U_{t,i}(a_k)$ to denote $UCB_{t,i}(a_k|\bar{\boldsymbol{o}}_{t,i})$ for any $a_k \in \mathcal{A}$ in this section. Expanding only the last term of the right-hand side of Eq. (24) yields:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{s\in S}(g_{i,s}(t)Q_{i,s}(t))|\boldsymbol{Q}_i(t)\right] \\
&= \mathbb{E}\left[\eta U_{t,i}(a^*_{t,i}) + \sum_{s\in a^*_{t,i}} Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right] - \mathbb{E}\left[\eta U_{t,i}(a^*_{t,i})|\boldsymbol{Q}_i(t)\right] \\
&\geq \mathbb{E}\left[\eta U_{t,i}(a^*_{t,i}) + \sum_{s\in a^*_{t,i}} Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right] - \eta r_{max}.
\end{aligned} \tag{25}$$

Substituting Eq. (25) into Eq. (24) gives

$$\begin{aligned}
&\Delta(L(\boldsymbol{Q}_i(t))) \\
&\leq \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) + \eta r_{max} \\
&\quad - \mathbb{E}\left[\eta U_{t,i}(a^*_{t,i}) + \sum_{s\in a^*_{t,i}} Q_{i,s}(t))|\boldsymbol{Q}_i(t)\right].
\end{aligned} \tag{26}$$

Note that the selection fraction vector $\boldsymbol{f}_i$ for any intersection i in our model should be strictly inside the maximal feasibility region $\mathcal{C}$. Therefore, there exists an $\epsilon > 0$ such that $\boldsymbol{f}_i + \epsilon\mathbf{1} = (f_{i,1} + \epsilon, \ldots, f_{i,|S|} + \epsilon)$ is also strictly inside the maximal feasibility region. Thus, there must exist an Ω-only policy $\boldsymbol{\phi}^{\Omega}_i$ that can support $\boldsymbol{f}_i + \epsilon\mathbf{1}$ according to lemma 1, which means policy $\boldsymbol{\phi}^{\Omega}_i$ can achieve the fairness constraints with the required minimum fraction $\boldsymbol{f}_i + \epsilon\mathbf{1}$:

$$\sum_{a_k\in\mathcal{A}} p^{\Omega}_{i,a_k}\mathbb{I}(s \in a_k) \geq f_{i,s} + \epsilon. \tag{27}$$

Here, $\mathbb{I}(s \in a_k)$ is an indicator function which equals 1 when $s \in a_k$. Since the FCTSC algorithm selects phases according to Eq. (19), we have:

$$\begin{aligned}
&\eta U_{t,i}(a^*_{t,i}) + \sum_{s\in a^*_{t,i}} Q_{i,s}(t)|\boldsymbol{Q}_i(t) \\
&\geq \eta U_{t,i}(a^{\Omega}_{t,i}) + \sum_{s\in a^{\Omega}_{t,i}} Q_{i,s}(t)|\boldsymbol{Q}_i(t), \forall a_k \in \mathcal{A},
\end{aligned} \tag{28}$$

where $a^{\Omega}_{t,i}$ is the chosen phase under policy $\boldsymbol{\phi}^{\Omega}_i$. Substituting Eq. (28) into Eq. (26) gives:

$$\begin{aligned}
&\Delta(L(\boldsymbol{Q}_i(t))) \\
&\leq \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) + \eta(r_{max} - r_{min}) \\
&\quad - \mathbb{E}\left[\sum_{s\in a^{\Omega}_{t,i}} Q_{i,s}(t)|\boldsymbol{Q}_i(t)\right] \\
&= \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) + \eta(r_{max} - r_{min}) \\
&\quad - \sum_{s\in S}(Q_{i,s}(t) \sum_{a_k\in\mathcal{A}} p^{\Omega}_{i,a_k}\mathbb{I}(s \in a_k)).
\end{aligned} \tag{29}$$

Applying Eq. (27) gives:

$$\begin{aligned}
&\Delta(L(\boldsymbol{Q}_i(t))) \\
&\leq \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) + \eta(r_{max} - r_{min}) \\
&\quad - \sum_{s\in S} Q_{i,s}(t)(f_{i,s} + \epsilon) \\
&= \frac{|S|}{2} + \eta(r_{max} - r_{min}) - \epsilon\sum_{s\in S} Q_{i,s}(t) \\
&= B - \epsilon\sum_{s\in S} Q_{i,s}(t),
\end{aligned} \tag{30}$$

where $B \stackrel{\text{def}}{=} \frac{|S|}{2} + \eta(r_{max} - r_{min})$.

Finally, by adopting the Lyapunov Drift Theorem [31], we have:

$$\lim_{T\to\inf}\frac{1}{T}\sum_{t=1}^{T}\sum_{s\in S_i}\mathbb{E}[Q_{i,s}(t)] \leq \frac{B}{\epsilon} < \infty. \tag{31}$$

Thus:

$$\lim_{T\to\inf}\frac{1}{T}\mathbb{E}[|Q_{i,s}(T)|] = 0, \forall s \in S. \tag{32}$$

Theorem 1 holds. □

5.2 Convergence Analysis

In this section, we undertake a comprehensive analysis of the convergence characteristics inherent in FCTSC algorithm, as defined by Eq. (11). We substantiate our investigation with the following theorems.

Theorem 2. The incomplete information multi-agent game in FCTSC converges to an asymptotically stable equilibrium at a rate that is at least $1/\sqrt{t}$ as $t \to \inf$ almost surely, i.e.,

$$P\left\{\lim_{t\to\inf}\max_{a_k\in\mathcal{A}}\left(\mu_i(a_k|\boldsymbol{o}_{t,i},\mathbb{P}_{t,i}) - \mu_i(a^*_{t,i}|\boldsymbol{o}_{t,i},\mathbb{P}_{t,i})\right) = 0\right\} = 1. \tag{33}$$

Proof. In [32], it is demonstrated that a perturbed game, where players employ fictitious play to formulate their strategies, and rewards are subject to stochastic fluctuations, converges to an asymptotically stable equilibrium with near certainty. It is worth noting that our model squarely fits within this category of perturbed games, since rewards are treated as random variables, each adhering to a specific probability distribution in the MAB framework. Consequently, we can confidently assert that the game in FCTSC converges almost certainly. Furthermore, by leveraging conditional limit theorems elucidated in studies such as those by Arthur [33] and Arthur [34], we substantiate that the rate of convergence satisfies or even surpasses the threshold of $1/\sqrt{t}$ as t tends towards infinity.

Theorem 2 holds. □

5.3 Regret

Before the detailed regret analysis, we first establish an optimal policy as a benchmark. Let us consider a scenario where the expected reward for any phase $a_k \in \mathcal{A}$ remains constant at μ_{a_k}. Then, we solve the following linear programming problem to find an optimal policy for intersection i that can

maximize the expected reward while still satisfying fairness constraints:

$$Maximize: \sum_{a_k \in \mathcal{A}} p_{i,a_k} \mu_{a_k} \tag{34}$$

$$Subject\ to: \sum_{s \in a_k} p_{i,a_k} \geq f_{i,s}, \forall s \in S, \tag{35}$$

$$\sum_{a_k \in \mathcal{A}} p_{i,a_k} = 1, \tag{36}$$

$$p_{i,a_k} \in [0,1], \forall a_k \in \mathcal{A}. \tag{37}$$

Clearly, the optimal solution to this linear programming problem can be considered as an Ω-policy, and we refer to it as an optimal Ω-policy.

Now, we can define the optimal policy $\boldsymbol{\phi}_i^{opt}$ as follows: at each round t, $\boldsymbol{\phi}_i^{opt}$ calculates the optimal Ω-policy only based on the current expected rewards, denoted by $\varphi_{t,i}^{opt}$, and then uses this policy to select a phase. That is to say, $\boldsymbol{\phi}_i^{opt} = \{\varphi_{1,i}^{opt}, ..., \varphi_{t,i}^{opt}, ..., \varphi_{T,i}^{opt}\}$. It can be easily demonstrated that $\boldsymbol{\phi}_i^{opt}$ ensures fairness. Let R^{opt} denote the accumulated reward under policy $\boldsymbol{\phi}_i^{opt}$. Then, the regret of FCTSC is defined as follows:

$$\mathcal{R}_F \stackrel{\text{def}}{=} R^{opt} - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{t,i}(a_{t,i}^*)\right]. \tag{38}$$

Based on the above definition, we can derive the bound about the regret of FCTSC as follows:

Theorem 3. Under the FCTSC algorithm, the regret defined in Eq. (38) has the following upper bound:

$$\mathcal{R}_F \leq \frac{|S|T}{2\eta} + C_2\sqrt{T}\log(TL) + C_3\sqrt{T}, \tag{39}$$

where $C_2, C_3 > 0$ are two suitably large constants and L adheres to the condition that $\|\boldsymbol{o}_{t,i}\|_2 \leq L$ for any observation $\boldsymbol{o}_{t,i}$.

Proof. We use $a_{t,i}^{opt}$ to denote the chosen phase in intersection i under policy $\boldsymbol{\phi}_i^{opt}$, and $g_{i,s}^{opt}(t)$ to indicate whether a movement signal s is selected. We can express the regret as follows:

$$\begin{aligned}\mathcal{R}_F &= R^{opt} - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{t,i}(a_{t,i}^*)\right] \\ &= \sum_{t=1}^{T} \mathbb{E}\left[\mu_{t,i}(a_{t,i}^{opt}) - \mu_{t,i}(a_{t,i}^*)\right].\end{aligned} \tag{40}$$

Moving forward, we use the drift-plus-penalty method to construct the following expression:

$$\begin{aligned}&L(\boldsymbol{Q}_i(t+1)) - L(\boldsymbol{Q}_i(t)) + \eta(\mu_{t,i}(a_{t,i}^{opt}) - \mu_{t,i}(a_{t,i}^*)) \\ &\leq \frac{|S|}{2} + \sum_{s\in S} f_{i,s}Q_{i,s}(t) - \sum_{s\in S} g_{i,s}(t)Q_{i,s}(t) \\ &\quad + \eta(\mu_{t,i}(a_{t,i}^{opt}) - \mu_{t,i}(a_{t,i}^*)) \\ &= \frac{|S|}{2} + \sum_{s\in S}(f_{i,s} - g_{i,s}^{opt}(t))Q_{i,s}(t) + \sum_{s\in a_{t,i}^{opt}} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^{opt}) \\ &\quad - \sum_{s\in a_{t,i}^*} Q_{i,s}(t) - \eta\mu_{t,i}(a_{t,i}^*).\end{aligned} \tag{41}$$

Taking expectations and summing for t from 1 to T on both sides yields:

$$\begin{aligned}&\mathbb{E}\left[L(\boldsymbol{Q}_i(T+1)) - L(\boldsymbol{Q}_i(1))\right] + \eta\mathcal{R}_F \\ &\leq \frac{|S|T}{2} + \sum_{t=1}^{T}\mathbb{E}\left[\left(\sum_{s\in a_{t,i}^{opt}} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^{opt})\right)\right. \\ &\quad \left. - \left(\sum_{s\in a_{t,i}^*} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^*)\right)\right],\end{aligned} \tag{42}$$

where the inequality holds because $\mathbb{E}\left[f_{i,s} - g_{i,s}^{opt}(t)\right] \leq 0$ due to the definition of policy $\boldsymbol{\phi}_i^{opt}$ and the properties of Ω-policies.

According to the definitions of $L(\boldsymbol{Q}_i(t))$ and $\boldsymbol{Q}_i(t)$, it is evident that $L(\boldsymbol{Q}_i(T+1)) \geq 0$ and $L(\boldsymbol{Q}_i(1)) = 0$. Therefore, we can conclude:

$$\begin{aligned}\mathcal{R}_F \leq \frac{|S|T}{2\eta} + \frac{1}{\eta}\sum_{t=1}^{T}\mathbb{E}&\left[\left(\sum_{s\in a_{t,i}^{opt}} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^{opt})\right)\right. \\ &\left. - \left(\sum_{s\in a_{t,i}^*} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^*)\right)\right].\end{aligned} \tag{43}$$

Now, let us consider an alternative policy $\boldsymbol{\phi}_i'$. This policy selects phases based on the following rule:

$$a_{t,i}' = \underset{a_k\in\mathcal{A}}{\operatorname{argmax}} \left(\eta\mu_{t,i}(a_k) + \sum_{s\in a_k} Q_{i,s}(t)\right). \tag{44}$$

Compared to policy $\boldsymbol{\phi}_i^{opt}$ and FCTSC, we can express:

$$\eta U_{t,i}(a_{t,i}^*) + \sum_{s\in a_{t,i}^*} Q_{i,s}(t) \geq \eta U_{t,i}(a_{t,i}') + \sum_{s\in a_{t,i}'} Q_{i,s}(t), \tag{45}$$

$$\eta\mu_{t,i}(a_{t,i}') + \sum_{s\in a_{t,i}'} Q_{i,s}(t) \geq \eta\mu_{t,i}(a_{t,i}^{opt}) + \sum_{s\in a_{t,i}^{opt}} Q_{i,s}(t). \tag{46}$$

Using Eq. (45) and Eq. (46), the expression inside the expectation in Eq. (43) can be further bounded as follows:

$$\begin{aligned}&\sum_{s\in a_{t,i}^{opt}} Q_{i,s}(t) + \eta\mu_{t,i}(a_{t,i}^{opt}) - \sum_{s\in a_{t,i}^*} Q_{i,s}(t) - \eta\mu_{t,i}(a_{t,i}^*) \\ &\leq \eta\left((U_i(a_{t,i}^*) - \mu_{t,i}(a_{t,i}^*)) + (\mu_{t,i}(a_{t,i}') - U_{t,i}(a_{t,i}'))\right).\end{aligned} \tag{47}$$

We employ $\bar{\boldsymbol{o}}_{t,i}^*$ to represent the expected observation that is computed by using the true probability distribution $\mathbb{P}_i$. As such, we have:

$$\begin{aligned}&(U_{t,i}(a_{t,i}^*) - \mu_{t,i}(a_{t,i}^*)) + (\mu_{t,i}(a_{t,i}') - U_{t,i}(a_{t,i}')) \\ =&(UCB_{t,i}(a_{t,i}^*|\bar{\boldsymbol{o}}_{t,i}) - \mu_{t,i}(a_{t,i}^*)) \\ &+ (\mu_{t,i}(a_{t,i}') - UCB_{t,i}(a_{t,i}'|\bar{\boldsymbol{o}}_{t,i})) \\ \leq&|UCB_{t,i}(a_{t,i}^*|\bar{\boldsymbol{o}}_{t,i}^*) - UCB_{t,i}(a_{t,i}^*|\bar{\boldsymbol{o}}_{t,i})| \\ &+ (UCB_{t,i}(a_{t,i}^*|\bar{\boldsymbol{o}}_{t,i}^*) - \mu_{t,i}(a_{t,i}^*)) \\ &+ |UCB_{t,i}(a_{t,i}'|\bar{\boldsymbol{o}}_{t,i}^*) - UCB_{t,i}(a_{t,i}'|\bar{\boldsymbol{o}}_{t,i})| \\ &+ (\mu_{t,i}(a_{t,i}') - UCB_{t,i}(a_{t,i}'|\bar{\boldsymbol{o}}_{t,i}^*)).\end{aligned} \tag{48}$$

Subsequently, by employing Theorem 2, we ascertain that:

$$\begin{aligned}&(U_{t,i}(a^*_{t,i}) - \mu_{t,i}(a^*_{t,i})) + (\mu_{t,i}(a'_{t,i}) - U_{t,i}(a'_{t,i}))\\ \leq&\frac{C_1}{\sqrt{t}} + (UCB_i(a^*_{t,i}|\bar{\boldsymbol{o}}^*_{t,i}) - \mu_{t,i}(a^*_{t,i}))\\ &+ (\mu_{t,i}(a'_{t,i}) - UCB_i(a'_{t,i}|\bar{\boldsymbol{o}}^*_{t,i})),\end{aligned} \tag{49}$$

where $C_1 > 0$ is a suitably large constant.

Moreover, drawing upon a similar analytical process to Chapter 19 in [35], we deduce:

$$\begin{aligned}&\sum_{t=1}^{T} \mathbb{E}[(UCB_{t,i}(a^*_{t,i}|\bar{\boldsymbol{o}}^*_{t,i}) - \mu_{t,i}(a^*_{t,i}))\\ &+ (\mu_{t,i}(a'_{t,i}) - UCB_{t,i}(a'_{t,i}|\bar{\boldsymbol{o}}^*_{t,i}))]\\ \leq&C_2\sqrt{T}\log(TL),\end{aligned} \tag{50}$$

where $C_2 > 0$ is a suitably large constant and L adheres to the condition that $\|\boldsymbol{o}_{t,i}\|_2 \leq L$ for any observation $\boldsymbol{o}_{t,i}$.

Substituting Eq. (47), Eq. (48), Eq. (49) and Eq. (50) into Eq. (43) gives:

$$\mathcal{R}_F \leq \frac{|S|T}{2\eta} + C_2\sqrt{T}\log(TL) + C_3\sqrt{T}, \tag{51}$$

where $C_3 > 0$ is a suitably large constant. The theorem holds. □

Through Eq.(51), we can discern the following breakdown: the first component represents the cost incurred for ensuring fairness, the second component reflects the regret stemming from reward estimation, and the third component signifies the cost necessary for the game in our algorithm to converge. Upon analyzing Eq.(51) and Eq. (19), it becomes evident that when η takes on a large value, the FCTSC algorithm prioritizes maximizing rewards over satisfying fairness constraints, leading to a smaller regret. Conversely, when η is small, the FCTSC algorithm tends to select phases with larger virtual queue lengths, which strengthens fairness guarantees but increases the regret of the algorithm.

6 Performance Evaluation

In this section, we meticulously evaluate the performance of the FCTSC algorithm through a series of simulations on real-world and synthetic datasets, including the average travel time, convergence speed, and fairness attributes.

6.1 Datasets and Settings

We conducted simulations using the CityFlow open-source traffic simulator [36]. In the simulator, vehicles are allowed to enter and exit from any edge in the network. Our simulations encompass both synthetic and real-world traffic flow datasets. For synthetic datasets, we configure Grid networks of different sizes: 1x3, 3x3, and 4x4. The arrivals of vehicles on each edge road of the network are assumed to follow a Poisson distribution with an approximate rate of 200 vehicles per hour. The road lengths and maximum speed of vehicles in these synthetic networks are fixed at 300 meters and 30km/h. The turning ratios at the intersections were set to 10% (left), 60% (straight), and 30% (right), which were based on statistical analyses of real-world traffic datasets, ensuring a realistic representation.

TABLE 2: Data statistics of real-world traffic dataset

Dataset	# of intersections	Arrival rate (vehicles/h)			
		mean	min	max	std
Hangzhou 4x4	16	2983	2400	4020	494.18
Manhattan 3x16	48	2824	2100	3660	321.41
Manhattan 7x28	196	10675	4740	13320	1623.64

For the real-world datasets, we incorporate real road networks from Hangzhou (4x4) and Manhattan (3x16 and 7x28) into the simulator, maintaining the authentic road lengths. To generate traffic flow data, we employ the open-source taxi trip data from LibSignal [4] which provides us with information on the starting times, origins, and destinations of every vehicle. The key statistics of the real-world traffic flow data are shown in Table 2.

6.2 Compared Algorithms and Evaluation Metrics

In this paper, we adopt the following widely-used four baseline algorithms for comparison:

- **FixedTime**: Fixedtime is a traditional traffic signal control method that sets a fixed time for each phase, regardless of traffic conditions.
- **MaxPressure** [3]: This is a classic state-of-the-art traffic signal control method, which greedily chooses the phase which has the maximum pressure.
- **IDQN** [4]: IDQN is an individual DRL-based approach, where each agent makes the traffic signal control decision only depending on its own intersection information without any cooperations.
- **PressLight** [5]: PressLight integrates pressure into the state and reward design for the DRL model, which autonomously achieves a certain extent of coordination across intersections without any prior knowledge.

We evaluate the performance of these algorithms and FCTSC with three metrics. The first is the **average travel time**, a straightforward metric that is defined as the average time taken by all vehicles to traverse the area. The second metric is the **convergence speed** which is actually the number of training rounds required for the algorithm to converge to stability. Finally, the last metric that we are concerned about is the **fairness**, indicated by using the average waiting time at each lane and the standard deviation of the average waiting time across all lanes.

6.3 Evaluation Results

6.3.1 Average Travel Time

Through the simulations on synthetic and real-world datasets, we obtain the average travel time of all vehicles under different configurations for each algorithm. The final statistics for the average travel time are presented in Table 3.

For synthetic data simulations, FCTSC, IDQN, and PressLight outperform the traditional algorithms FixedTime and MaxPressure. In the Grid1x3 dataset, FCTSC performs slightly worse than IDQN and PressLight, while in the Grid3x3 and Grid4x4 datasets, FCTSC's performance is

TABLE 3: Average travel time on synthetic data and real-world data

Model	Average travel time (s)					
	Grid1x3	Grid3x3	Grid4x4	Hangzhou4x4	Manhattan3x16	Manhattan7x28
Fixedtime	128.8	231.9	300.3	575.5	1122.9	1582.1
MaxPressure	99.2	161.7	199.5	369.6	282.8	1300.9
IDQN	**94.2**	156.1	192.1	351.9	217.9	1266.9
PressLight	95.0	156.0	193.9	352.1	222.3	1252.7
FCTSC	97.9	**155.3**	**189.9**	**351.7**	**212.9**	**1207.6**

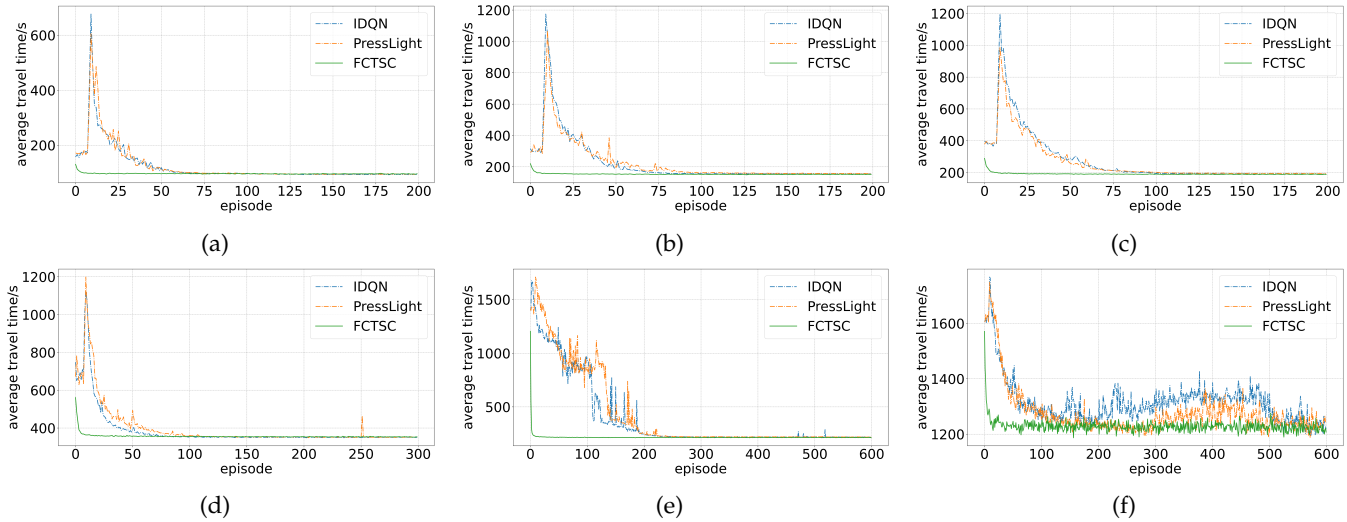

Fig. 3: Convergence speed of FCTSC, IDQN and PressLight in different traffic setting. (a) Grid1x3 (b) Grid3x3 (c) Grid4x4 (d) Hangzhou4x4 (e) Manhattan3x16 (f) Manhattan7x28

TABLE 4: Average waiting times and corresponding standard deviations for different algorithms.

	Average waiting time (s)			
	MaxPressure	IDQN	PressLight	FCTSC
Lane 1	1717.6	1858.0	1858.0	204.7
Lane 2	212.2	3482.0	3482.1	398.0
Lane 3	199.5	17.9	16.4	231.9
Lane 4	204.0	17.9	16.4	213.4
Lane 5	199.5	3482.0	16.7	231.9
Lane 6	203.9	3482.0	16.7	213.4
Lane 7	191.3	16.4	3482.0	198.2
Lane 8	1931.8	16.4	3482.0	393.3
std	753.9	1719.3	1719.6	**84.2**

better than IDQN and PressLight. This is attributed to the fact that the road network of synthetic dataset Grid1x3 is relatively small, making FCTSC have no space to leverage its effectiveness in cooperation between intersections. Additionally, FCTSC, while optimizing vehicle travel time, also needs to consider fairness and convergence speed, contributing to its suboptimal performance in this scenario.

When transitioning to real-world data simulations, FCTSC demonstrates performance that surpasses all traditional and DRL-based algorithms. As seen in Table 3, this trend becomes more pronounced for increasingly complex road networks. Notably, on the most complex 7x28 road network, FCTSC reduces average travel time by 4.68% compared to IDQN and 3.60% compared to PressLight. This highlights that our FCTSC algorithm effectively leverages cooperation between intersections in complex road networks, confirming the algorithm's efficiency.

6.3.2 *Convergence Speed*

We gather the average travel time of every training episode for each algorithm and visualize the results in Figure 3. This graphical representation emphasizes that in every simulation configuration, FCTSC consistently achieves significantly faster convergence compared to the other two RL-based methods, PressLight and IDQN. While FCTSC converges in fewer than 50 episodes, PressLight and IDQN require hundreds of episodes to reach convergence. Notably, in Figure 3f, both DRL-based algorithms, IDQN and PressLight, experience a prolonged period of performance instability before convergence, and in Figure 3b, the IDQN algorithm experiences a notable decline in performance after approximately 600 episodes, whereas our FCTSC algorithm maintains stability across all scenarios. This proves that FCTSC demonstrates outstanding performance in terms of convergence speed and stability.

6.3.3 *Fairness*

To evaluate the fairness of these algorithms, a simulation was conducted with significantly different traffic volumes in various lanes. Specifically, the simulation involves eight

lanes, with seven lanes having a vehicle arrival rate of one vehicle every six seconds, while one lane has a vehicle arrival rate of one vehicle every 120 seconds. In a scenario where fairness is not considered, the average waiting times for vehicles in different lanes would exhibit substantial disparities. Table 4 presents the results of the average waiting times for vehicles in each lane, along with the standard deviation of the waiting times for each algorithm.

From the table, it is evident that the MaxPressure, IDQN, and PressLight algorithms yield substantial imbalances in the average waiting times of vehicles across different lanes. The standard deviation of the traditional MaxPressure algorithm is 753.9, while the two DRL-based algorithms, IDQN and PressLight, have standard deviations of over 1700. Clearly, in this scenario, IDQN and PressLight would result in some lanes with lower traffic volume having no opportunities for passage. In contrast, FCTSC achieves a much more equitable distribution of average waiting times with only a standard deviation of 84.2, effectively mitigating disparities among lanes. These results underscore that FCTSC successfully addresses fairness concerns, resulting in a more equitable distribution of traffic flow.

7 Conclusion

In this paper, we introduce an online traffic signal control problem, where the fairness and the cooperation between neighboring intersections are taken into consideration simultaneously, and we model this problem as a contextual MAB issue with an incomplete information multi-agent game and a fairness constraint. Then, we propose the FCTSC algorithm to solve the problem, in which a novel UCB index is designed to address the contextual bandit issue with the fairness constraint and the fictitious play is employed to solve the incomplete information game issue. Moreover, we prove the fairness and the convergence the equilibrium of FCTSC, and derive an upper bound on the regret.

References

[1] Matthew Beedham, "What does TomTom Traffic Index data tell us about the world's busiest cities?" 2023, https://www.tomtom.com/newsroom/explainers-and-insights/the-most-congested-cities-in-the-world-2022/.

[2] J. McBride, N. Berman, and A. Siripurapu, "The state of U.S. infrastructure," 2023, https://www.cfr.org/backgrounder/state-us-infrastructure.

[3] H. Wei, G. Zheng, V. Gayah, and Z. Li, "A survey on traffic signal control methods," *arXiv preprint arXiv:1904.08117*, 2019.

[4] H. Mei, X. L. Lei, L. Da, B. Shi, and H. Wei, "LibSignal: An Open Library for Traffic Signal Control," in *Proc. NeurIPS RL4RL Workshop*, 2022.

[5] H. Wei, C. Chen, G. Zheng, K. Wu, V. Gayah, K. Xu, and Z. Li, "Presslight: Learning max pressure control to coordinate traffic signals in arterial network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2019, pp. 1290–1298.

[6] Y. Chen, S. Zhang, Y. Yan, Y. Jin, N. Chen, M. Ji, and M. Xiao, "Crowd2: Multi-agent bandit-based dispatch for video analytics upon crowdsourcing," in *Proc. IEEE INFOCOM*, 2023, pp. 1–10.

[7] H. Wang, Y. Yang, E. Wang, W. Liu, Y. Xu, and J. Wu, "Truthful user recruitment for cooperative crowdsensing task: A combinatorial multi-armed bandit approach," *IEEE Trans. Mobile Comput.*, vol. 22, no. 7, pp. 4314–4331, 2023.

[8] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, "Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2779–2794, 2021.

[9] H. Wang, Y. Yang, E. Wang, W. Liu, Y. Xu, and J. Wu, "Combinatorial multi-armed bandit based user recruitment in mobile crowdsensing," in *Proc. IEEE SECON 2020*, pp. 1–9.

[10] Q. Kang and W. P. Tay, "Task recommendation in crowdsourcing based on learning preferences and reliabilities," *IEEE Trans. Serv. Comput.*, vol. 15, no. 4, pp. 1785–1798, 2022.

[11] G. Gao, S. Huang, H. Huang, M. Xiao, J. Wu, Y.-E. Sun, and S. Zhang, "Combination of auction theory and multi-armed bandits: Model, algorithm, and application," *IEEE Trans. Mobile Comput.*, vol. 22, no. 11, pp. 6343–6357, 2023.

[12] G. Gao, H. Huang, M. Xiao, J. Wu, Y.-E. Sun, and S. Zhang, "Auction-based combinatorial multi-armed bandit mechanisms with strategic arms," in *Proc. IEEE INFOCOM*, 2021, pp. 1–10.

[13] L. Lyu, J. Yu, K. Nandakumar, Y. Li, X. Ma, J. Jin, H. Yu, and K. S. Ng, "Towards fair and privacy-preserving federated deep models," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 11, pp. 2524–2541, 2020.

[14] L. Zhang, Z. Wang, X. Dong, Y. Feng, X. Pang, Z. Zhang, and K. Ren, "Towards Fairness-aware Adversarial Network Pruning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 5168–5177.

[15] Z. Wang, X. Dong, H. Xue, Z. Zhang, W. Chiu, T. Wei, and K. Ren, "Fairness-aware adversarial perturbation towards bias mitigation for deployed deep models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10 379–10 388.

[16] X. Pang, D. Guo, Z. Wang, P. Sun, and L. Zhang, "Towards fair and efficient task allocation in blockchain-based crowdsourcing," *CCF Trans. Netw.*, vol. 3, pp. 193–204, 2020.

[17] W. Wang, Y. Yang, Z. Yin, K. Dev, X. Zhou, X. Li, N. M. F. Qureshi, and C. Su, "BSIF: Blockchain-Based Secure, Interactive, and Fair Mobile Crowdsensing," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3452–3469, 2022.

[18] X. Zhu, J. An, M. Yang, L. Xiang, Q. Yang, and X. Gui, "A Fair Incentive Mechanism for Crowdsourcing in Crowd Sensing," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1364–1372, 2016.

[19] B. P. Gokulan and D. Srinivasan, "Distributed geometric fuzzy multiagent urban traffic signal control," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 714–727, 2010.

[20] D. Teodorović, "Swarm intelligence systems for transportation engineering: Principles and applications," *Transp. Res. Part C: Emerg. Technol.*, vol. 16, no. 6, pp. 651–667, 2008.

[21] S. El-Tantawy, B. Abdulhai, and H. Abdelgawad, "Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1140–1150, 2013.

[22] H. Wei, G. Zheng, H. Yao, and Z. Li, "IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2018, p. 2496–2505.

[23] G. Zheng, Y. Xiong, X. Zang, J. Feng, H. Wei, H. Zhang, Y. Li, K. Xu, and Z. Li, "Learning Phase Competition for Traffic Signal Control," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, 2019, p. 1963–1972.

[24] T. Nishi, K. Otaki, K. Hayakawa, and T. Yoshimura, "Traffic signal control based on reinforcement learning with graph convolutional neural nets," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.* IEEE, 2018, pp. 877–883.

[25] T. Chu, J. Wang, L. Codecà, and Z. Li, "Multi-agent deep reinforcement learning for large-scale traffic signal control," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1086–1095, 2019.

[26] H. Wei, N. Xu, H. Zhang, G. Zheng, X. Zang, C. Chen, W. Zhang, Y. Zhu, K. Xu, and Z. Li, "Colight: Learning network-level cooperation for traffic signal control," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, 2019, pp. 1913–1922.

[27] Y. Xiong, G. Zheng, K. Xu, and Z. Li, "Learning traffic signal control from demonstrations," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, 2019, pp. 2289–2292.

[28] L. Prashanth and S. Bhatnagar, "Reinforcement learning with average cost for adaptive control of traffic lights at intersections," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst.*, 2011, pp. 1640–1645.

[29] D. Ghosh and C. Knapp, "Estimation of traffic variables using a linear model of traffic flow," *Transp. Res.*, vol. 12, no. 6, pp. 395–402, 1978.

[30] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A Contextual-Bandit Approach to Personalized News Article Recommendation," in *Proc. 19th Int. Conf. World Wide Web*, 2010, p. 661–670.

[31] M. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Springer Nature, 2010.
[32] M. Benaım and M. W. Hirsch, "Mixed equilibria and dynamical systems arising from fictitious play in perturbed games," *Games Econ. Behav.*, vol. 29, no. 1-2, pp. 36–72, 1999.
[33] V. B. Artur, Yu. M. Ermol'ev, and Yu. M. Kaniovskii, "Adaptive Growth Processes Modeled by URN Schemes," *Cybernetics*, vol. 23, no. 6, pp. 779–789, 1987.
[34] W. B. Arthur, Y. M. Ermoliev, and Y. M. Kaniovski, "Nonlinear adaptive processes of growth with general increments: Attainable and unattainable components of terminal set," 1988.
[35] T. Lattimore and C. Szepesvári, *Bandit Algorithms*. Cambridge University Press, 2020.
[36] H. Zhang, S. Feng, C. Liu, Y. Ding, Y. Zhu, Z. Zhou, W. Zhang, Y. Yu, H. Jin, and Z. Li, "CityFlow: A Multi-Agent Reinforcement Learning Environment for Large Scale City Traffic Scenario," in *Proc. Int. World Wide Web Conf.*, 2019, p. 3620–3624.

Junjie Shao is currently a master's student at the University of Science and Technology of China. He received his B.E. degree from the University of Science and Technology of China in 2021. His research interests include edge computing, game theory, and intelligent transportation.

Yu Zhao is currently a master in University of Science and Technology of China. He received his bachelor degree from Chongqing University in 2021. His research interests include edge computing, autonomous driving and reinforcement learning.

Jinbo Cai is currently a master's student at the University of Science and Technology of China. He received his B.E. degree from Wuhan University in 2021. His research interests edge computing, mobile crowdsensing and intelligent transportation.

He Sun received his B.S. degree from the School of Computer Science and Technology and B.A. degree from the School of Foreign Languages, Qingdao University, Qingdao, China in 2020. He is currently pursuing the Ph.D. degree on computer science with the School of Computer Science and Technology, University of Science and Technology of China (USTC), Hefei, China. His research interests include reinforcement learning, game theory, data collection&trading, and privacy preservation.

Jie Wu is Laura H. Carnell Professor at Temple University and the Director of the Center for Networked Computing (CNC). He served as Chair of the Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and Associate Vice Provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University, where he received his Ph.D. in 1989. His current research interests include mobile computing and wireless networks, routing protocols, network trust and security, distributed algorithms, applied machine learning, and cloud computing. Dr. Wu regularly published in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Service Computing and Journal of Computer Science and Technology. Dr. Wu is/was general chair/co-chair for IEEE DCOSS'09, IEEE ICDCS'13, ICPP'16, IEEE CNS'16, WiOpt'21, ICDCN'22, IEEE IPDPS'23, and ACM MobiHoc'23 as well as program chair/cochair for IEEE MASS'04, IEEE INFOCOM'11, CCF CNCC'13, and ICCCN'20. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a Fellow of the AAAS and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award. He is a Member of the Academia Europaea (MAE).

Mingjun Xiao (Member, IEEE) received the PhD degree from the University of Science and Technology of China in 2004. He is currently a professor with the School of Computer Science and Technology, University of Science and Technology of China. He has authored or coauthored more than 100 papers in referred journals and conferences, including TSC, TMC, TC, TPDS, TON, TKDE, INFOCOM, and ICDE, etc. His research interests include mobile crowdsensing, edge computing, federated learning, auction theory, data security and privacy. He was a TPC member of INFOCOM'23, INFOCOM'22, IJCAI'22, INFOCOM'21, IJCAI'21, and so on. He is on the reviewer board of several top journals, such as TSC, TMC, TON, TPDS, TVT, and TC.