Urban Traffic Light Control via Active Multi-Agent Communication and Supply-Demand Modeling

Xin Guo, Zhengxu Yu, Pengfei Wang, Zhongming Jin, Jianqiang Huang, Deng Cai, *Senior Member, IEEE*, Xiaofei He, *Senior Member, IEEE*, and Xian-Sheng Hua, *Fellow, IEEE*

Abstract—Urban traffic light control is an important and challenging real-world problem. By regarding intersections as agents, most of the reinforcement learning-based methods generate agents' actions independently. They can cause action conflict and result in overflow or road resource waste in adjacent intersections. Recently, some collaborative methods have alleviated the above problems by extending the observable surroundings of agents, which can be considered inactive cross-agent communication methods. However, when agents act synchronously in these works, the perceived action value is biased, and the information exchanged is insufficient. In this work, we first propose a novel Multi-agent Communication and Action Rectification (MaCAR) framework. It enables active communication between agents by considering the impact of synchronous actions of agents. Another fundamental problem of traffic light control is the balance between traffic demand and road supply capacity. To fully describe the relation between traffic demand and road supply capacity (Supply-Demand modeling, SD), we further model and forecast the Supply-Demand relation to facilitating the effectiveness of the model's action. The experiments show that our model outperforms state-of-the-art methods on both synthetic and real-world datasets. Combining the SD with MaCAR, SD-MaCAR can further boost the traffic light control performance even in traffic accident scenarios.

Index Terms—Traffic light control, reinforcement learning, supply-demand modeling

1 Introduction

Urban Traffic Light Control (TLC) is a critical and challenging real-world problem that maximizes traffic efficiency with limited urban road resources and avoids traffic conflict inside intersections. Finding an appropriate TLC approach can significantly mitigate traffic congestion and bring in significant economic, environmental and societal benefits [1].

Most of the recently proposed reinforcement learning (RL) based methods [1], [2], [3] are focusing on independently controlling intersections in the same road network. In these works, the observable surrounding of an agent is limited to itself, which may cause action conflict between agents and deviate action value from expectation. Increasing green-light time in directions suffering heavy traffic can significantly mitigate the traffic. However, it may cause severe congestion in adjacent regions if the adjacent agents cannot adjust themselves in time. Such congestion often occurs because traffic changes caused by agent action changes can be rapid and abrupt.

To overcome this problem, some collaborative optimization-based works [2], [4], [5], [6] are proposed recently. These works mitigate the action conflict problem mainly by expanding the observable surroundings of agents. It can be seen as establishing an inactive communication mechanism between agents, in which agents can obtain not only the traffic state of themselves but also adjacent intersections. This mechanism is inactive because agents do not directly share decision information such as actions and agent states but share noised responses of the road network.

There are two shortcomings of this inactive communication mechanism. First, agents cannot know each other's new actions before perceiving the traffic changes in this inactive communication mechanism. Hence, the perceived action values of agents are biased since agents perceive traffic pattern changes constantly lag behind action change. Second, the information propagated is insufficient for collaborative optimization because the shared information only includes the surrounding traffic states but does not include important agent decision information such as historical actions.

In this work, we propose a novel Multi-agent Communication and Action Rectification (MaCAR) framework, consisting of two parts. The first part is a Communication Agent Network (CAN) with a Message Propagation Graph Neural Network (MPGNN) based active communication

- *Xin Guo, Deng Cai, and Xiaofei He are with the State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang 310058, China. E-mail: guoxinzju@gmail.com, {dengcai, xiaofeihe}@cad.zju.edu.cn.*
- *Zhengxu Yu, Zhongming Jin, Jianqiang Huang, and Xian-Sheng Hua are with DAMO Academy, Alibaba Group, Hangzhou, Zhejiang 311121, China. E-mail: {yuzxfred, huaxiansheng}@gmail.com, {zhongming.jinzm, jianqiang.hjq}@alibaba-inc.com.*
- *Pengfei Wang is with DAMO Academy, Alibaba Group, Hangzhou, Zhejiang 311121, China, and also with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China. E-mail: wpf228618@alibaba-inc.com.*

Manuscript received 14 April 2021; revised 8 October 2021; accepted 21 November 2021. Date of publication 24 November 2021; date of current version 7 March 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0101400, in part by the National Nature Science Foundation of China under Grants 62036009, U1909203, 61936006, and 62133013, in part by Innovation Capability Support Program of Shaanxi under Grant 2021TD-05.
(Corresponding author: Deng Cai.)
Recommended for acceptance by M. A. Cheema.
Digital Object Identifier no. 10.1109/TKDE.2021.3130258

network. The purpose of CAN is to generate actions considering other agents' actions and states. By introducing the action information, agents can effectively learn the traffic pattern under the corresponding action, reducing the action space and improving model performance. The second part of MaCAR is a novel Traffic Forecasting Network (TFN). TFN forecast the traffic of the whole road network under agents' new actions, together with the corresponding action values. Using this predicted information, we further amend the action value of agents during training to mitigate action value deviation.

Moreover, we are the first to combine supply-demand modeling and forecasting with TLC tasks to help facilitate the effectiveness of the TLC model's actions in this work. The fundamental problem of TLC is the balance between traffic demand and road supply capacity. A variety of transport problems originating from the contradiction between travel demand and road supply capacity have been discussed in the relational literature [7], [8]. Supply-demand (SD) relation modeling and forecasting are extensively involved in traffic domains to fully describe the relationship between travel demand and road supply capacity. As an extension of MaCAR, we further apply supply-demand modeling with MaCAR (SD-MaCAR) to help facilitate the TLC performance.

On the traffic demand side, we model and forecast the historical traffic state indexes like the queue length of each intersection. The queue length of each intersection indicates how many vehicles will enter a subsequent road. Hence, we can use the queue length as an index of incoming traffic of the following road. On the road supply side, we model and forecast indexes like variation of traffic flow of the target road in a time window. The road supply capacity reflects the traffic volume the target road can bear in a specific time period. Modeling the road supply capacity can help restrain the model's action in an acceptable scope. Moreover, it could also reveal possible traffic accidents that happened on the target road. Hence, supply-demand modeling can help improve the model's performance under traffic accident scenarios.

We carry out extensive experiments on both synthetic and real-world datasets to evaluate the performance of MaCAR and SD-MaCAR. Experimental results demonstrate that by introducing the active communication mechanism, taking advantage of traffic forecasting information, and considering the supply-demand (SD) relation, our proposed model can achieve superior performance against state-of-the-art methods.

We summarize the contributions of this work as follows:

1) We propose a novel Multi-agent Communication and Action Rectification (MaCAR) framework to control multiple agents collaboratively. MaCAR consists of two components: (1) a novel Message Propagation based Communication Agent Network (CAN), which generates coordinated actions via an active cross-agent communication mechanism; (2) a novel Traffic Forecasting Network (TFN), which helps to rectify action against action value bias.
2) As an extension of MaCAR, we further involve supply-demand modeling with MaCAR (SD-MaCAR) in this work to help boost the model's performance. Specifically, we model and forecast each intersection's queue length and traffic flow variation as indexes of demand and supply dimension, respectively.
3) In experiments, we demonstrate that our proposed method can achieve state-of-the-art performance on both synthetic and real-world datasets by involving the active communication mechanism and taking advantage of traffic forecasting information. Moreover, the experimental results indicate our SD-MaCAR model can outperform all baselines in various traffic scenarios by involving supply-demand modeling.

In the following sections, we summarize the recent works related to reinforcement learning-based traffic light control, Graph Neural Network, and Supply-Demand Modeling and Forecasting in Section 2. We introduce our MaCAR and SD-MaCAR model in Sections 3 and 4 respectively. Finally, we give the conclusion in Section 6.

2 Related Works

2.1 Reinforcement Learning Based Traffic Light Control

2.1.1 Independent Control Versus Collaborative Control

Most recently proposed methods [1], [9], [10], [11] are focusing on optimizing agent independently, in which an independent RL agent is compelled to control a certain intersection without noticing other agents' existing. However, as part of the whole road network, the impact of other agents cannot be completely blocked out in real-world applications.

To mitigate this problem, some collaborative control based methods are proposed recently [4], [6], [12], [13]. Wei *et al.* [6] proposes the CoLight, an independent TLC algorithm with inactive communication mechanism. Their inactive communication mechanism extends the agent's receptive field to help make coordinated action. Agents do not share decision information directly in their work but rather share the road network's responses to historical actions. Therefore, agents do not know the latest actions of other agents until it feels the traffic changes, resulting in a possible action conflict. Different from them, MaCAR is a multi-agent collaborative control-based method with an active communication mechanism. In MaCAR, we use the traffic prediction information to actively rectify agents' future actions during training by alleviating action value bias caused by other agents' actions.

2.1.2 Adaptive Control Versus Pre-Defined Phase Scheme

Most recently proposed RL methods [1], [2], [3], [6], [9], [14] are based on an adaptive control strategy in which the agent selects whether switch the permitted phase (lanes) per second based on the traffic situation. Adaptive control provides excellent controllability since the permitted phase can be switched in seconds. However, frequent phase switching may cause traffic accidents and significantly affect the driving experience. To avoid bad driving experiences and traffic accidents caused by a sudden phase switching, some predefined phase scheme-based RL approaches [3], [9] are proposed. A typical phase scheme consists of a fixed phase execution cycle and the corresponding execution time list.

The agent's action is to adjust the phase execution time. Similarly, MaCAR uses a pre-defined phase scheme-based control strategy to provides practicability in real-world applications.

2.2 Graph Neural Network

Recently proposed GNNs can be divided into two categories, spectral-based (mainly GCN based) methods [15], [16], and Message Propagation based methods [17], [18]. A typical GCN-based model trained on a specific graph can not be directly applied to a graph with a different graph [19]. Hence, plain GCNs is inconvenient in a TLC scenario because multiple agents' actions can significantly change each road's weight. More recently, several Message Propagation based methods have been proposed to solve this problem [18]. Similarly, MaCAR uses Message Propagation based neural networks to model dynamic graphs from traffic.

2.3 Supply-Demand Modeling and Forecasting

Two major and parallel approaches exist to identify a transportation system's critical elements in the transportation literature. On the one hand, conventional transportation engineering emphasizes travel demand, often in terms of traffic volume (i.e., demand-side). On the other hand, newer techniques from Network Science emphasize network topology (i.e., supply-side) [8]. Both demand and supply sides are indispensable in order to provide a full description of the traffic states. Hence, supply-demand modeling and forecasting are widely used in previous transportation literature [7], [20], [21], [22]. [7] establishes an estimation model of urban transportation supply-demand ratio (TSDR) to quantitatively describe the conditions of an urban transport system and to support a theoretical basis for transport policy-making. [20] proposes to tackle the large-scale fleet management problem using a contextual multi-agent reinforcement learning framework. To optimize the Wireless Charging Lane (WCL) deployment in urban areas, [22] focuses on installation cost reduction while achieving regional balance of energy supply and demand, as well as vehicle continuous operability issues.

In this work, we are the first to combine supply-demand modeling and forecasting with the TLC tasks to help facilitate the TLC model's actions.

3 MaCAR

We first introduce some notations and terms. We define term *period* as the time length of executing all phases once. For simplicity, we set the period length of all intersections as P to ensure agents' synchronous actions. The action applied in period t is defined as $\mathrm{a}^{(t)}$, and its corresponding perceived action value is defined as $\mathrm{v}^{(t)}$. The adjacent intersection set of intersection n is defined by $\mathcal{N}(n)$, and it is obtained from the road network topology G.

The input traffic state $\mathrm{q}^{(t)}$ is a tensor that contains the queue lengths of all roads in the road network after tth period. Our purpose is to minimize the queue length $\mathrm{q}^{(t)}$ of the whole road network, which is a common practice in previous works. To achieve that, we use a concatenated tensor contains the input traffic state and the corresponding actions of the past M periods as input, and we denote this concatenated tensor as $\{\{\mathrm{q},\mathrm{a}\}^{(t-\mathrm{M})},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\}$, where M is a training interval hyper-parameter selected by practice.

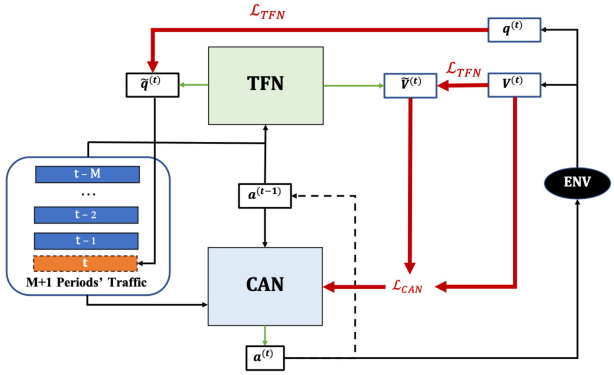


Fig. 1. Framework of the multi-agent communication and action rectification (MaCAR).

3.1 Communication Agent Network

The framework of MaCAR has shown in Fig. 1, the first part is the Communication Agent Network (CAN). CAN is a multi-agent action generation network consisting of an MPGNN based central active communication network and several customized agent subnets, as shown in Fig. 2.

3.1.1 Active Communication Network

The active communication network of CAN is formed by several linear layers and Message Propagation based Graph Neural Networks (MPGNNs), which is designed by following the strong discriminant graph theory proved by Xu *et al.* [23]. MPGNN learns the spatio-temporal message propagation mechanism between agents.

MPGNN has a multi-layered network architecture, which can learn complex propagation patterns with a wide receptive field. Formally, the feature embedding of intersection n in kth layer $\mathrm{h}_n^{\mathrm{k}}$is

$$\mathrm{m}_n^{\mathrm{k}} = f_p\left(\sum_{u\in\mathcal{N}(n)} \mathrm{h}_u^{\mathrm{k}-1}\right), \tag{1}$$

$$\mathrm{h}_n^{\mathrm{k}} = f_a((1+\epsilon^{\mathrm{k}})\mathrm{h}_n^{\mathrm{k}-1} + \mathrm{m}_n^{\mathrm{k}}), \tag{2}$$

where $\mathrm{m}_n^{\mathrm{k}}$ is the aggregated message received from adjacent intersections of intersection n in kth layer. f_p and f_a is the propagation and aggregation function respectively, both of them are learned by using a neural network consists of two linear layers. ϵ^{k} is a learnable parameter which helps to generate discriminant graph representation. The input intersection features $\{\{\mathrm{q}_n,\mathrm{a}_n\}^{(t-\mathrm{M})},\ldots,\{\mathrm{q}_n,\mathrm{a}_n\}^{(t-1)}\}$ are defined as h_n^0. The output of the last layer of MPGNN $\mathrm{h}^{\mathrm{out}}$ is a graph embedding contains all aggregated features $\{\mathrm{h}_n^{\mathrm{out}} \mid n \in G\}$.

Based on MPGNN, we establish the active communication network to learn the message aggregation and propagation among agents under certain actions, as shown in Fig. 2. We then feed the output into agent subnets to generate actions.

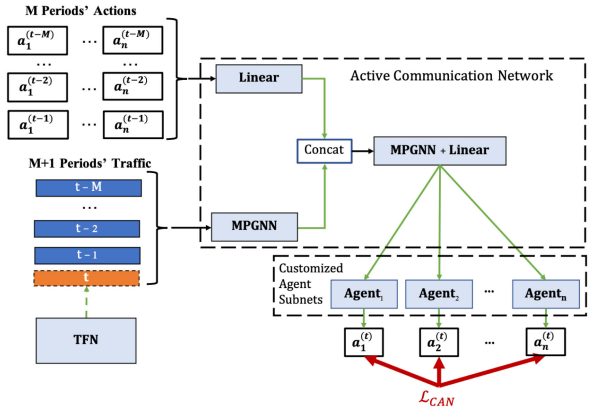


Fig. 2. Sketch of the communication agent network (CAN).

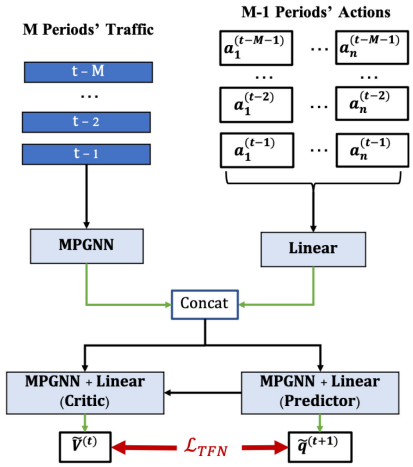


Fig. 3. Sketch of the traffic forecasting network (TFN).

3.1.2 Customized Agent Subnet

To fit different types of intersection's traffic light settings, agents in CAN are still decentralized agents. The intersection number determines the number of customized agent subnets. Meanwhile, the agent subnet is customized according to the corresponding intersection settings, such as its channelization or direction number. By doing so, MaCAR can fit most existing intersection settings and commonly used control strategies.

Each agent (subnet) of CAN learns a continuous action distribution $\mathcal{P}$. We then sample action from the learned distribution using the following equations

$$Sample\ \mathrm{a}_n^{(t)} \sim \mathcal{P}_n(f_n(\mathrm{h}^{\mathrm{out}})), \tag{3}$$

$$\mathrm{a}_n^{(t)} = softmax(\alpha\ \mathrm{a}_n^{(t)}), \tag{4}$$

where n is the intersection index, f_n learns the mapping from the global feature embedding $\mathrm{h}^{\mathrm{out}}$ to the mean and variance of action distribution $\mathcal{P}_n$.

Action sampled from the continuous distribution $\mathcal{P}$ is first uniformed by using a uniform multiplier $\alpha > 0$, and then rescaled by using a Softmax function to generate the new phase executing time ratio $\mathrm{a}_n^{(t)}$. At last, we multiply $\mathrm{a}_n^{(t)}$ and P to get the new execution time list.

3.2 Traffic Forecasting Network

The second part of MaCAR is the Traffic Forecasting Network (TFN), which aims to predict the future traffic and the action value of the given actions. We amend the action value and rectify agent action against other agents' actions using this forecasting information.

As shown in Fig. 3, TFN first extracts feature embeddings from the previous M periods' traffic using an MPGNN (k=2), and then concatenates them with the embeddings of the corresponding actions. The concatenated embeddings are then fed into two branches (Predictor and Critic). The Predictor branch is used to predict the future traffic $\widetilde{\mathrm{q}}^{(t)}$ of the whole road network. The Critic branch aims to predict the action value $\widetilde{\mathrm{v}}^{(t)}$ of the given actions $\mathrm{a}^{(t)}$.

As shown in Fig. 3, there is a shortcut from the Predictor branch to the Critic branch. We argue that this shortcut can bring in the predicted future traffic under the given actions to help mitigate the impact of other agents' actions on $\widetilde{\mathrm{v}}^{(t)}$.

Algorithm 1. Online Training Algorithm for MaCAR

Input: Initial action $\mathrm{a}^{(0)}$, parameters θ_{TFN} and θ_{CAN}, period length P, simulation time length t_{max}, training interval M, simulator $\mathcal{S}$

Output: Optimized θ_{TFN} and θ_{CAN}

Let $t = \mathrm{M}$
run $\mathcal{S}(\mathrm{a}^{(0)})$ M periods $\rightarrow \{\{\mathrm{q},\mathrm{a}\}^{(0)},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\}$
$\mathrm{TFN}(\{\{\mathrm{q},\mathrm{a}\}^{(0)},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\};\theta_{\mathrm{TFN}}) \rightarrow \widetilde{\mathrm{v}}^{(t)},\widetilde{\mathrm{q}}^{(t)}$
$\mathrm{CAN}(\{\{\mathrm{q},\mathrm{a}\}^{(0)},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\},\widetilde{\mathrm{q}}^{(t)};\theta_{\mathrm{CAN}}) \rightarrow \mathrm{a}^{(t)}$
$\mathcal{S}(\mathrm{a}^{(t)}) \rightarrow \{\mathrm{q},\mathrm{a}\}^{(t)}$
$t = t + 1$
while $(t \times P < t_{max})$ **do**
  if $(t\ \%\ \mathrm{M} \neq 0)$ **then**
    $\mathrm{TFN}(\{\{\mathrm{q},\mathrm{a}\}^{(t-\mathrm{M})},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\};\theta_{\mathrm{TFN}}) \rightarrow \widetilde{\mathrm{v}}^{(t)},\widetilde{\mathrm{q}}^{(t)}$
    $\mathrm{CAN}(\{\{\mathrm{q},\mathrm{a}\}^{(t-\mathrm{M})},\ldots,\{\mathrm{q},\mathrm{a}\}^{(t-1)}\},\widetilde{\mathrm{q}}^{(t)};\theta_{\mathrm{CAN}}) \rightarrow \mathrm{a}^{(t)}$
    $\mathcal{S}(\mathrm{a}^{(t)}) \rightarrow \{\mathrm{q},\mathrm{a}\}^{(t)}$
    $t = t + 1$
  else
    calculate $\{\mathrm{v}^{(t-\mathrm{M})},\ldots,\mathrm{v}^{(t)}\}$ via Eq. (5)
    optimize $\theta_{\mathrm{TFN}},\theta_{\mathrm{CAN}}$ by minimizing Eqs. (6) and Eq. (7)
  end if
end while
return $\theta_{\mathrm{TFN}},\theta_{\mathrm{CAN}}$

3.3 Online Training Algorithm

We devise a policy gradient-based online training algorithm to train the CAN and TFN after each specific period M. The online training algorithm has shown in Algorithm 1.

As shown in Algorithm 1, to collect the initial information for MaCAR, we first let the simulator run for M periods under initial action $\mathrm{a}^{(0)}$ in which all phases have the same executing time. After that, new predictions and actions are generated using the traffic states and corresponding actions of previous M periods. We train the MaCAR network after every M period, in which one period is equal to P time steps. During training, we first calculate the action values of the past M periods by using Eq. (5). The action values of tth

period are defined by the difference of queue lengths between $t-1$ and t period

$$\mathrm{v}^{(t)} = \mathrm{q}^{(t-1)} - \mathrm{q}^{(t)}. \tag{5}$$

We then optimize the TFN network by minimizing the following loss function

$$\mathcal{L}_{\mathrm{TFN}} = \sum_{t=1}^{\mathrm{M}} (||\mathrm{v}^{(t)} - \widetilde{\mathrm{v}}^{(t)}||_{\ell_1} + ||\mathrm{q}^{(t)} - \widetilde{\mathrm{q}}^{(t)}||_{\ell_1}), \tag{6}$$

where $||\cdot||_{\ell_1}$ represents ℓ_1 norm. It's worth noting that both $\widetilde{\mathrm{v}}^{(t)}$ and $\widetilde{\mathrm{q}}^{(t)}$ are generated before generating $\mathrm{a}^{(t)}$.

As discussed above, other agents' newly generated actions will impact the action values and cause deviation between perceived and real action values. We use the difference between predicted and perceived action value to anchor agent action to overcome this problem. We optimize the CAN by minimizing the following loss function

$$\mathcal{L}_{\mathrm{CAN}} = \sum_{t=1}^{\mathrm{M}} \log(\mathrm{a}^{(t)})(\widetilde{\mathrm{v}}^{(t)} - \mathrm{v}^{(t)}). \tag{7}$$

4 Supply-Demand Based MaCAR

In a multitude of papers, travel demand and road supply capacity has been widely used to analyze traffic problems to clarify the theoretical and technical basis for efficient transport [8]. A variety of transport problems originating from the contradiction between travel demand and road supply capacity have been discussed in the relational literature [7], [8]. In the urban TLC scenario, effective modeling and forecasting future travel demand and road supply capacity are required in many aspects. It can improve the TLC model's effectiveness and effectively prevent the congestion caused by the imbalance between travel demand and road supply capacity.

To clearly illustrate the Supply-Demand relation in traffic light control, we give the definition of *supply* in Definition 1 and *demand* in Definition 2, respectively.

Definition 1 Supply *in a certain time window is defined as the road capacity, i.e., the remaining road resource to carry vehicles in this period. Specially, the* Future Supply *is the road capacity in the next time window.*

Definition 2 Demand *in certain time window is defined as the traffic condition, i.e., the occupied road resources by vehicles in this period. Specially, the* Future Demand *is the traffic condition in the next time window.*

Concretely, in our setting, we use the variation of vehicle number $v^{(t)}$ to represent the supply $\mathrm{s}^{(t)}$ as Eq. (8) and the queue length $q_i^{(t)}$ in each direction i of each lane to represent the demand $\mathrm{d}^{(t)}$ as Eq. (9). The demand in SD-MaCAR is distinguished from the M Periods' traffic in MaCAR. In the SD-MaCAR model, fine-grained lane information of each direction is considered instead of coarse-grained road states to represent the whole traffic graph's local original-destination (OD) information. Considering these local ODs, the dynamic global OD graph is implicitly assembled in our SD-MaCAR model.

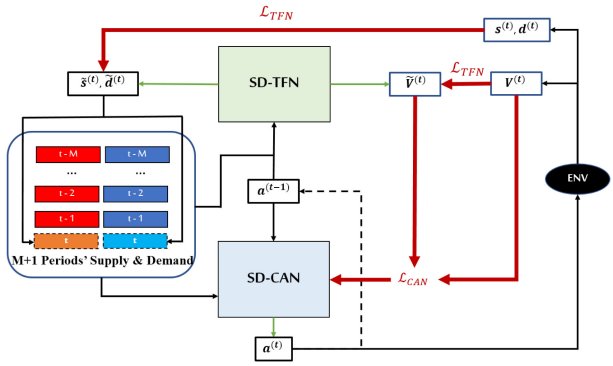


Fig. 4. Framework of the supply-demand based multi-agent communication and action rectification (SD-MaCAR).

$$\mathrm{s}^{(t)} = v^{(t)} - v^{(t-1)}. \tag{8}$$

$$\mathrm{d}^{(t)} = \{q_i^{(t)}\}_{i=1}^{I}. \tag{9}$$

This section further proposes an extension of MaCAR with the supply-demand modeling and forecasting module (SD-MaCAR) to handle both travel demand and road supply capacity information. Specifically, we replaced the original MPGNN and Linear layers in CAN and TFN networks with a newly proposed supply-demand module (a.k.a SD module). In the SD module, we retained the modeling of the demand side state as in MaCAR and integrated the road supply states of the road network to help the model learn when and how the traffic light should be adjusted.

4.1 Supply-Demand CAN

As shown in Fig. 4, we name the CAN with SD module as SD-CAN, and the architecture of SD-CAN has shown in Fig. 5. Compared with the original CAN, SD-CAN takes not only past M periods' actions and demand data as input but also road supply data as well.

As shown in Fig. 5, we reserved the MPGNN, which aims to model the past M periods' traffic, which indicates the historical travel demand. Moreover, we added another MPGNN to model the road supply data to fully describe the road network. After that, the embedding of past M periods' action, together with the past M periods' supply data and the predicted tth supply data from SD-TFN, are concatenated as the final supply feature embedding. The embedding of past M periods' action, the past M periods' demand data, and the predicted tth demand data from SD-TFN are concatenated as final demand features. Finally, the supply and demand features are fed into the following MPGNN and customized agent sub-nets.

4.2 Supply-Demand TFN

As shown in Fig. 6, similar to SD-CAN, we use two MPGNN to get the embedding of supply data and the feature of demand data in SD-TFN. The embedding of past M periods' action and the supply of the past M periods are concatenated as the final supply feature. The embedding of past M periods' action and the past M periods' demand are concatenated as the final demand feature. The supply and demand features are concatenated and feed into two MPGNN to predict the tth

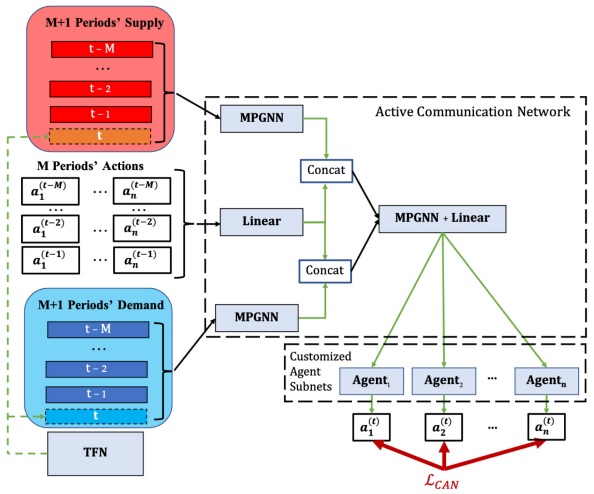


Fig. 5. Sketch of the supply-demand based communication agent network (SD-CAN).

supply and demand. Further, the predicted demand is used to predict the tth action value.

4.3 Incorporating With Accident Detection

In actual traffic conditions, the accident is essential information that cannot be omitted. We believe that a TLC model with the ability to detect the accident will be more robust when facing accident conditions. However, it is challenging to train an accident detection module and an online TLC model simultaneously. In our SD-MaCAR model, we adapt the pretrain-then-finetune strategy. Concretely, we first pretrain the supply model in SD-TFN offline, remove the last full-connect layer and use the pretrained model as the supply feature extractor. Finally, we jointly train the whole SD-MaCAR, in which the supply feature extractor is finetuned with a small learning rate.

4.4 Online Training Algorithm for SD-MaCAR Model

We also devise a policy gradient-based online training algorithm for SD-MaCAR, which is shown in Algorithm 2.

To collect the initial information for SD-MaCAR, simulators are first run for M periods under initial action $\mathrm{a}^{(0)}$ in which all phases have the same executing time. After that, new predictions for supply $\widetilde{\mathrm{s}}^{(t)}$, demand $\widetilde{\mathrm{d}}^{(t)}$ and actions $\widetilde{\mathrm{v}}^{(t)}$ are generated by using the previous supply and demand information and corresponding actions of previous M periods. We train the SD-MaCAR network after every M period, in which one period is equal to P time steps. During training, we first calculate the action values of the past M periods by using Eq. (10). The action values of tth period are defined by the difference of demand between $t-1$ and t period

$$\mathrm{v}^{(t)} = \mathrm{d}^{(t-1)} - \mathrm{d}^{(t)}. \tag{10}$$

We then optimize the SD-TFN network by minimizing the following loss function

$$\mathcal{L}_{\mathrm{TFN}} = \sum_{t=1}^{\mathrm{M}} (||\mathrm{v}^{(t)} - \widetilde{\mathrm{v}}^{(t)}||_{\ell_1} + ||\mathrm{s}^{(t)} - \widetilde{\mathrm{s}}^{(t)}||_{\ell_1} + ||\mathrm{d}^{(t)} - \widetilde{\mathrm{d}}^{(t)}||_{\ell_1}), \tag{11}$$

where $||\cdot||_{\ell_1}$ represents ℓ_1 norm. It's worth noting that both $\widetilde{\mathrm{v}}^{(t)}$, $\widetilde{\mathrm{s}}^{(t)}$, and $\widetilde{\mathrm{d}}^{(t)}$ are generated before generating $\mathrm{a}^{(t)}$.

As discussed above, other agents' newly generated actions will impact the action values and cause deviation between perceived and real action values. To overcome this problem, same as the training strategy of MaCAR, we use the difference between predicted and perceived action value to help to anchor agent action by using Eq. (7).

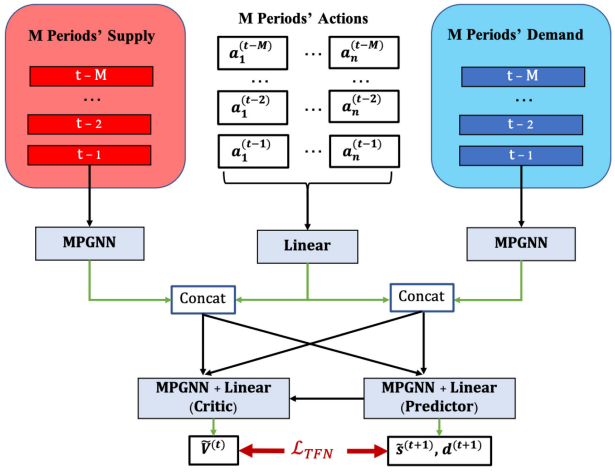


Fig. 6. Sketch of the supply-demand based traffic forecasting network (SD-TFN)

Algorithm 2. Online Training Algorithm for SD-MaCAR

Input: Initial action $\mathrm{a}^{(0)}$, parameters $\theta_{\mathrm{SD-TFN}}$ and $\theta_{\mathrm{SD-CAN}}$, period length P, simulation time length t_{max}, training interval M, simulator $\mathcal{S}$

Output: Optimized $\theta_{\mathrm{SD-TFN}}$ and $\theta_{\mathrm{SD-CAN}}$

Let $t = \mathrm{M}$
run $\mathcal{S}(\mathrm{a}^{(0)})$ M periods $\rightarrow \{\{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(0)}, \ldots, \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t-1)}\}$
$\mathrm{TFN}(\{\{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(0)}, \ldots, \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t-1)}\}; \theta_{\mathrm{SD-TFN}}) \rightarrow \widetilde{\mathrm{v}}^{(t)}, \widetilde{\mathrm{s}}^{(t)}, \widetilde{\mathrm{d}}^{(t)}$
$\mathrm{CAN}(\{\{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(0)}, \ldots, \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t-1)}\}, \widetilde{\mathrm{s}}^{(t)}, \widetilde{\mathrm{d}}^{(t)}; \theta_{\mathrm{SD-CAN}}) \rightarrow \mathrm{a}^{(t)}$
$\mathcal{S}(\mathrm{a}^{(t)}) \rightarrow \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t)}$
$t = t + 1$
while $(t \times P < t_{max})$ **do**
  if $(t \ \% \ \mathrm{M} \neq 0)$ **then**
    $\mathrm{TFN}(\{\{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t-\mathrm{M})}, \ldots, \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{t-1}\}; \theta_{\mathrm{SD-TFN}}) \rightarrow \widetilde{\mathrm{v}}^{(t)}, \widetilde{\mathrm{s}}^{(t)}, \widetilde{\mathrm{d}}^{(t)}$
    $\mathrm{CAN}(\{\{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t-\mathrm{M})}, \ldots, \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{t-1}\}, \widetilde{\mathrm{s}}^{(t)}, \widetilde{\mathrm{d}}^{(t)}; \theta_{\mathrm{SD-CAN}}) \rightarrow \mathrm{a}^{(t)}$
    $\mathcal{S}(\mathrm{a}^{(t)}) \rightarrow \{\mathrm{s},\mathrm{d},\mathrm{a}\}^{(t)}$
    $t = t + 1$
  else
    calculate $\{\mathrm{v}^{(t-\mathrm{M})}, \ldots, \mathrm{v}^{(t)}\}$ via Eq. (10)
    optimize $\theta_{\mathrm{SD-TFN}}, \theta_{\mathrm{SD-CAN}}$ by minimizing Eqs. (11) and Eq. (7)
  end if
end while
return $\theta_{\mathrm{SD-TFN}}, \theta_{\mathrm{SD-CAN}}$

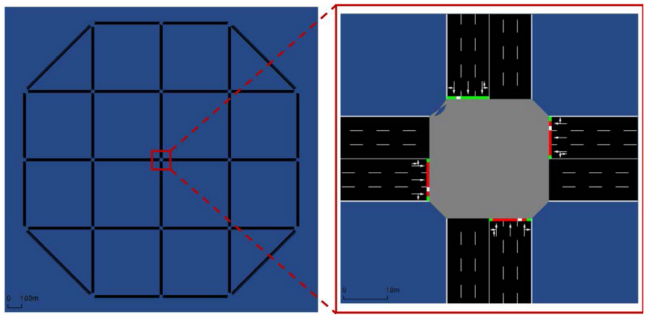


Fig. 7. Left: Sketch of the simulated 21 intersections road network [18] we used in synthetic experiments. Right: An intersection example with four directions.

TABLE 1
Configurations for Simulation Traffic

Config	Traffic	Arrival Rate (car/300s)	Trend	Start Time	End Time
1	T1	150	mixed	0	20000
	T2	150	mixed	0	20000
	T3	150	mixed	0	20000
2	T4	120	mixed	0	10800
	T5	180	mixed	10800	18000
	T6	300	mixed	10800	18000
3	T7	150	mixed	0	20000
	T8	210	W→E	0	20000
Accident	T9	330	mixed	0	5000

"W→E" represents that most vehicles are running from west to east on the road network.

5 Experiments

We conduct experiments in two open-source traffic simulators: CityFlow[1] and SUMO[2]. Both of them are commonly used in previous works [1], [6], [14].

5.1 Datasets

5.1.1 Real-World Datasets

We used three real-world datasets: $D_{Hangzhou}$, D_{Jinan} and $D_{NewYork}$. All three datasets are obtained from CoLight [6]. The New York, Hangzhou, and Jinan datasets have 196, 16, and 12 intersections, respectively. We used the same simulator (CityFlow) and experimental settings as CoLight to conducted a fair comparison. The data statistic of the used real-world traffic datasets has shown in Fig. 2.

5.1.2 Synthetic Datasets

In the SUMO simulator, we build a road network with 21 intersections to simulate the urban trunk roads, as shown in Fig. 7. The road network is constructed by following Wei *et al*. [18]. Intersections in the middle of the road network have four directions, and intersections on the boundary have three directions. Each road is 500 meters long, except roads in the corner are 680 meters. Each road has two directions and three lanes per direction. The road speed limit is set as 16.68 m/s, and the period length is 120 seconds.

As shown in Table 1, we built four different traffic scenarios based on the road network above to mimic the prevailing and challenging traffic conditions in the real world. In Config 1, we synthesized the daytime with heavy flat traffic. In Config 2, we synthesized the traffic switching from light flat-time traffic to heavy peak-time traffic. In Config 3, we simulated the tidal traffic in which part of traffic has a strong trend (West→East) during heavy peak-time traffic. In Config Accident, we simulated a car accident on lane B0C0 during the 1800 to 2800 seconds, which resulted in a traffic jam. All four kinds of configurations are widespread in metropolitan areas around the world.

The arrival rate defines the traffic density in these configurations. The traffic trend of different traffic groups (T1-T9 in Table 1) is determined by each road's probability of becoming a starting point via point and destination from a uniform distribution. End time represents the departure time of the last vehicle. All data contain bidirectional and dynamic flows with turning traffic. Moreover, we used the vehicle rerouter algorithm provided by SUMO to synthesize vehicle rerouting cases as in the real world.

5.1.3 Traffic Forecasting Dataset

To evaluate the traffic forecasting performance of TFN, we conduct experiments in the persuasive real-world traffic forecasting dataset METR-LA [15] comparing with several state-of-the-art methods.

5.2 Compared Methods

We compare MaCAR and SD-MaCAR with several baseline methods, including several state-of-the-art methods.

In synthetic experiments, we compare with four different baseline methods, including Fixed-time Control (FT), Random Adjustment (RA), Actuated Control (SOTL [24]), and Q-Learning Traffic Light Optimization within Multiple Intersections Traffic Network (QLTSO) [25]. Each baseline represents a type of commonly used method in real-world applications.

We then conduct experiments in three real-world datasets, comparing them with several state-of-the-art methods [1], [2], [4], [5], [6], [26]. The state-of-the-art method CoLight [6] is a collaborative optimization-based method, in which information exchange between agents is carried out by expanding other agents' observable surroundings.

5.3 Experiments on Synthetic Datasets

We first compare our proposed model with several baselines under four traffic scenarios. Experimental results have shown in Tables 3, 4, 5, and 6. We can notice that MaCAR

TABLE 2
Data Statistic of the Used Real-World Traffic Datasets [6]

Dataset	# Intersections	Arrival Rate (vehicles/300s)			
		Mean	Std.	Max	Min
$D_{NewYork}$	196	240.79	10.08	274	216
$D_{Hangzhou}$	16	526.63	86.70	676	256
D_{Jinan}	12	250.70	38.21	335	208

1. https://cityflow-project.github.io
2. http://sumo.dlr.de/index.html

TABLE 3
Results on Synthetic Dataset Config 1

Methods	Avg.Speed (m/s)	Avg.Queue	Avg.Waiting (s)
FT	4.22	3.08	28.55
RA	5.61	3.19	27.52
SOTL	6.76	2.44	17.64
QLTSO	5.35	3.71	41.76
MaCAR-noTFN	6.62	2.75	17.98
MaCAR	7.30	2.19	13.40
MaCAR+Supply	7.44	2.16	13.24
MaCAR+Demand	7.52	2.11	12.92
SD-MaCAR	**7.63**	**1.99**	**12.73**

TABLE 5
Results on Synthetic Dataset Config 3

Methods	Avg.Speed (m/s)	Avg.Queue	Avg.Waiting (s)
FT	3.47	4.72	50.88
RA	3.26	4.92	52.46
SOTL	3.18	4.33	47.32
QLTSO	7.25	2.15	15.83
MaCAR-noTFN	7.24	2.66	20.63
MaCAR	7.83	2.07	15.58
MaCAR+Supply	7.87	1.93	14.65
MaCAR+Demand	7.99	1.71	13.12
SD-MaCAR	**8.29**	**1.36**	**11.57**

can effectively improve traffic efficiency under all four different traffic conditions. Meanwhile, our proposed model outperforms all baselines significantly on Avg.Waiting in Config 2, Config 3, and Config Accident. It demonstrates that our method can still effectively change the phase plans to mitigate traffic congestion when traffic shifts.

Further experiments in Tables 3, 4, 5, and 6 shows that, the supply and demand information have positive influence to TLC. Our SD-MaCAR model outperforms all the other methods without supply or demand. Table 6 shows that our model performs quite well when facing accidents and traffic jams. Especially, combine our model with accident detection, SD-MaCAR* outperforms all of the other models.

In Figs. 8a and 8b, we show that MaCAR can significantly reduce the total queue length and increase the Avg.Speed. In Fig. 8b, we can notice the curve of MaCAR is flatter than other methods and has an upward trend even when the traffic starts to increase from 60 periods. Moreover, the upward trend of MaCAR's curve is more significant than MaCAR-noTFN. These experiments validate that our idea of taking advantage of prediction information is feasible while demonstrating the effectiveness of the proposed cross-agent communication mechanism. Also, the curve of SD-MaCAR is better than those that do not consider supply or demand information.

In Fig. 9, we learn that our MaCAR related models significantly outperform other baselines. Especially, SD-MaCAR is the most robust model facing an accident, which shows the necessity of using supply and demand information.

Fig. 10 shows the study for accident detection. We can learn that the accident detection models (MaCAR+Supply* and SD-MaCAR*) have a larger average speed, a lower average queue number, and a lower average waiting time. These experiments tell us that accident detection is important in TLC when an accident occurs.

5.4 Experiments on Real-World Datasets

We then compare our proposed model with several state-of-the-art methods in real-world datasets. We followed the experimental setup used by CoLight to compare our method with previous works fairly. Results have shown in Table 7, MaCAR achieves consistent performance improvements over state-of-the-art methods on all three real-world datasets: the average improvement is 2.78% compare with CoLight. The performance improvements of our method are attributed to the unique design of our proposed model.

Our method also outperforms the joint-action modeling method CGRL. To achieve cooperation, CGRL establishes one model to determine the joint actions of two adjacent intersections and then conducts centralized coordination of the global joint actions. It needs to search an ample operating space and may face scalability issues. Compared with CGRL, our method has a smaller search space since we still use independent agents.

Experiments on the New York dataset show that MaCAR can outperform state-of-the-arts in a 196 intersections road network. These experiments also demonstrate that MaCAR has good scalability in practice. Our SD-MaCAR can further improve the experiment result on these real-world datasets, proving the importance of supply-demand modeling in the actual task.

TABLE 4
Results on Synthetic Dataset Config 2

Methods	Avg.Speed (m/s)	Avg.Queue	Avg.Waiting (s)
FT	7.82	0.81	6.94
RA	8.20	0.81	6.75
SOTL	8.42	0.70	5.32
QLTSO	8.13	0.95	10.19
MaCAR-noTFN	8.29	0.71	5.43
MaCAR	8.69	0.68	5.13
MaCAR+Supply	8.95	0.64	4.11
MaCAR+Demand	9.13	0.59	4.15
SD-MaCAR	**9.33**	**0.61**	**3.85**

TABLE 6
Results on Synthetic Dataset Config Accident

Methods	Ave.Speed (m/s)	Avg.Queue	Avg.Waiting(s)
FT	3.08	2.25	28.25
RA	4.51	1.95	20.05
SOTL	4.12	1.90	18.46
QLTSOL	3.50	2.47	32.49
MaCAR	5.58	1.60	14.18
MaCAR+Demand	5.86	1.52	12.95
MaCAR+Supply	5.77	1.53	13.64
MaCAR+Supply*	5.84	1.51	12.93
SD-MaCAR	5.87	1.46	12.78
SD-MaCAR*	**5.93**	**1.41**	**12.41**

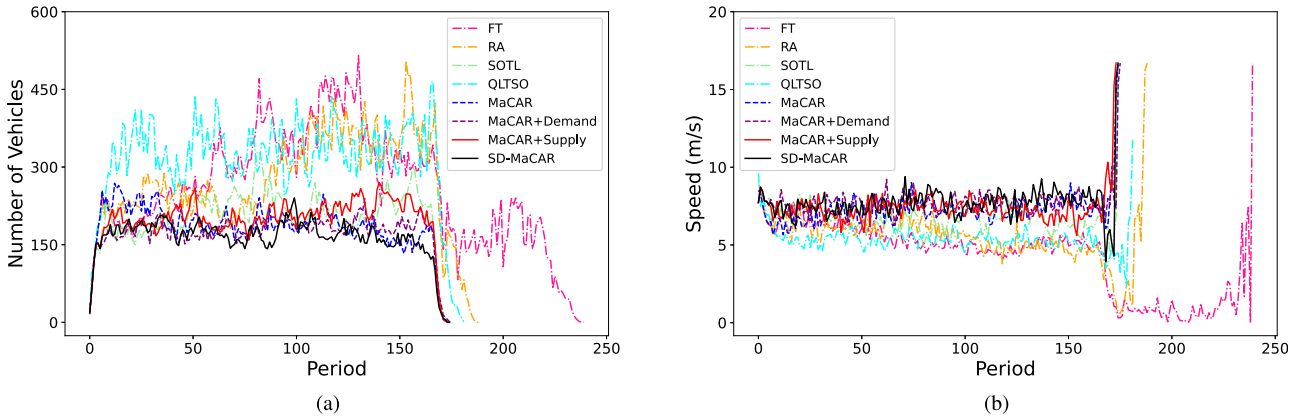


Fig. 8. Traffic config 1. (a) Total queue length of the road network. The lower the curve, the higher the road patency. (b) The average speed of all vehicles. The higher the curve, the higher the road network efficiency.

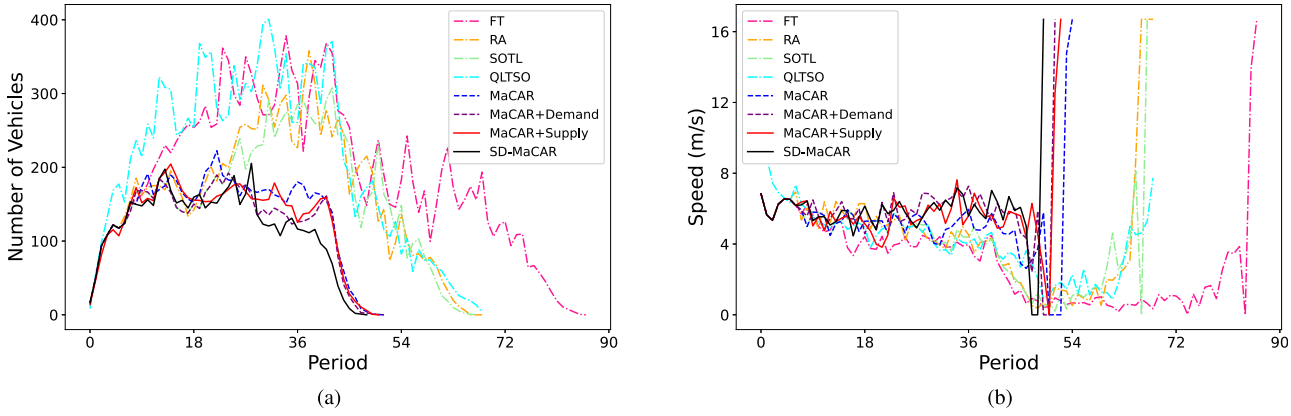


Fig. 9. Traffic config accident. (a) Total queue length of the road network. The lower the curve, the higher the road patency. (b) The average speed of all vehicles. The higher the curve, the higher the road network efficiency.

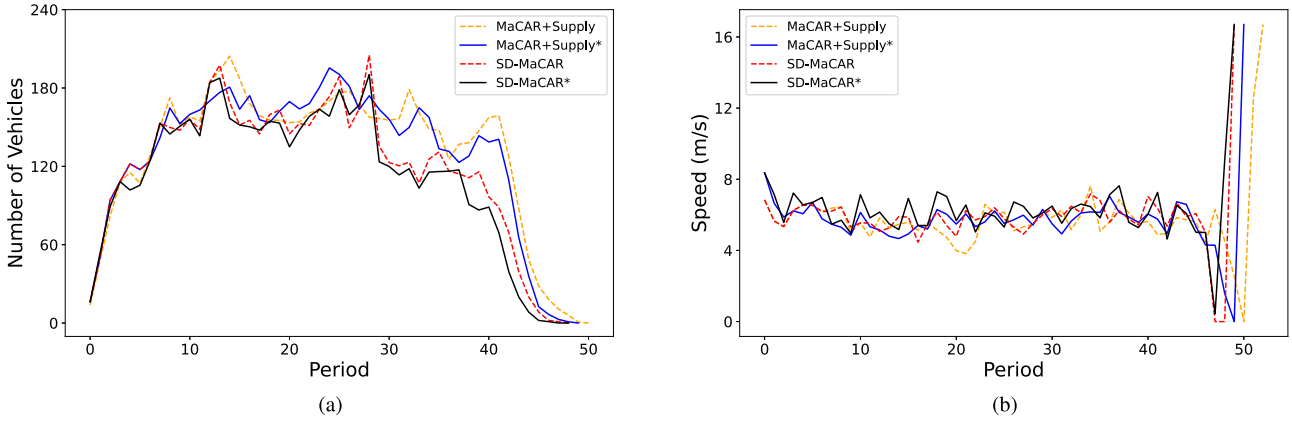


Fig. 10. Results for accident-detection pretraining under traffic config accident. (a) Total queue length of the road network. The lower the curve, the higher the road patency. (b) The averagespeed of all vehicles. The higher the curve, the higher the road network efficiency.

5.5 Experiments on Traffic Forecasting Dataset

We then conduct traffic forecasting experiments on the METR-LA dataset to demonstrate the traffic forecasting performance of TFN comparing with state-of-the-art traffic forecasting methods. Results have shown in Table 8, TFN and SD-TFN can outperform all the state-of-the-art methods. We can notice that the state-of-the-art methods may perform appropriately for short-term forecasting, but their long-term predictions are not accurate due to the error accumulation.

5.6 Ablation Study

We also compared the performance of variants of our proposed method, as shown in Tables 3, 4, 5, and 6. By removing TFN, we notice that the performance of MaCAR-noTFN is reduced significantly but still can outperform baseline methods in most metrics. These experiments show that our idea of combining traffic forecasting with TLC can effectively improve traffic efficiency.

We can also learn from these tables that, MaCAR+Supply and MaCAR+Demand outperform our basic MaCAR model,

TABLE 7
Results on Real-World Datasets w.r.t Average Travel Time

Methods	$D_{NewYork}$	$D_{Hangzhou}$	D_{Jinan}
CGRL [4]	2187.12	1582.25	1210.70
NeighborRL [26]	2280.92	1053.45	1168.32
GCN [2]	1876.37	768.43	625.66
OneModel [5]	1973.11	394.56	728.63
Individual RL [1]	-	345.00	325.56
CoLight [6]	1459.28	297.26	291.14
MaCAR	1425.00 (+2.30%)	291.18 (+2.04%)	279.49 (+4.00%)
SD-MaCAR	**1422.65 (+2.51%)**	**290.69 (+2.21%)**	**278.94 (+4.19%)**

We show the improvement of both MaCAR and SD-MaCAR compared with CoLight.

TABLE 8
Traffic Forecasting Experiment Results on METR-LA

Methods	MAE		
	15mins	30mins	60mins
FC-LSTM	3.44	3.77	4.37
STGCN [27]	2.87	3.48	4.45
DCRNN [15]	2.77	3.15	3.60
ST-UNet [16]	2.72	3.12	3.55
TFN	2.68	2.99	3.28
SD-TFN	**2.53**	**2.78**	**3.11**

MAE metric is compared for different future time steps.

and the SD-MaCAR, which combines the supply and demand, is even better. This experiment shows that supply and demand are essential in the SD-MaCAR model.

6 Conclusion

In this work, we address the traffic light control problem by proposing a novel Multi-agent Communication and Action Rectification (MaCAR) framework. As an extension of MaCAR, we further involve supply-demand modeling with MaCAR (SD-MaCAR) in this work to help boost the model's performance. We conduct extensive experiments on synthetic and real-world datasets to demonstrate the superior performance of our proposed method over baseline and state-of-the-art methods. Besides, we show in-depth case studies and observations to understand how the proposed method overcomes two shortcomings of previous collaborative optimization methods. We also prove the significance of using the Supply-Demand relation in traffic light control. Several future directions are worth exploring, global optimization methods that consider local importance and causal inference-based methods.

Acknowledgment

Xin Guo and Zhengxu Yu contributed equally.

References

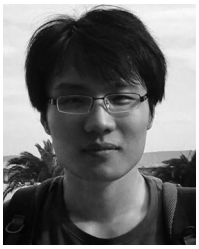
[1] H. Wei, G. Zheng, H. Yao, and Z. Li, "Intellilight: A reinforcement learning approach for intelligent traffic light control," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 2496–2505.
[2] T. Nishi, K. Otaki, K. Hayakawa, and T. Yoshimura, "Traffic signal control based on reinforcement learning with graph convolutional neural nets," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 877–883.
[3] N. Casas, "Deep deterministic policy gradient for urban traffic light control," 2017, *arXiv:1703.09035*.
[4] E. Van der Pol and F. A. Oliehoek, "Coordinated deep reinforcement learners for traffic light control," in *Proc. Workshop Learn. Inference Control Multi-Agent Syst.*, 2016.
[5] 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, Mar. 2020.
[6] H. Wei *et al.*, "CoLight: Learning network-level cooperation for traffic signal control," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, 2019, pp. 1913–1922.
[7] C. Wu, Y. Pei, and J. Gao, "Model for estimation urban transportation supply-demand ratio," *Math. Problems Eng.*, vol. 2015, pp. 1–12, 2015.
[8] M. Akbarzadeh, S. Memarmontazerin, S. Derrible, and S. F. S. Reihani, "The role of travel demand and network centrality on the connectivity and resilience of an urban street system," *Transportation*, vol. 46, no. 4, pp. 1127–1141, 2019.
[9] M. Aslani, M. S. Mesgari, and M. Wiering, "Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events," *Transp. Res. Part C: Emerg. Technol.*, vol. 85, pp. 732–752, 2017.
[10] G. Zheng *et al.*, "Diagnosing reinforcement learning for traffic signal control," 2019, *arXiv:1905.04716*.
[11] C. Chen *et al.*, "Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 3414–3421.
[12] D. Lee, N. He, P. Kamalaruban, and V. Cevher, "Optimization for reinforcement learning: From single agent to cooperative agents," in *IEEE Signal Process. Mag.*, vol. 37, no. 3, 2020, pp. 123–135.
[13] H. Zhang, C. Liu, W. Zhang, G. Zheng, and Y. Yu, "Generalight: Improving environment generalization of traffic signal control via meta reinforcement learning," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, 2020, pp. 1783–1792.
[14] H. Wei *et al.* "Presslight: Learning max pressure control to coordinate traffic signals in arterial network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 1290–1298.
[15] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Representations*, 2018.
[16] B. Yu, H. Yin, and Z. Zhu, "ST-UNet: A spatio-temporal u-network for graph-structured time series modeling," 2019, *arXiv:1903.05631*.
[17] P. W. Battaglia *et al.*, "Relational inductive biases, deep learning, and graph networks," 2018, *arXiv:1806.01261*.
[18] L. Wei *et al.*, "Dual graph for traffic forecasting," *IEEE Access*, early access, Dec. 09, 2019, doi: 10.1109/ACCESS.2019.2958380.
[19] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.
[20] K. Lin, R. Zhao, Z. Xu, and J. Zhou, "Efficient collaborative multi-agent deep reinforcement learning for large-scale fleet management," 2018, *arXiv:1802.06444*.
[21] B. Amini, F. Peiravian, M. Mojarradi, and S. Derrible, "Comparative analysis of traffic performance of urban transportation systems," *Transp. Res. Rec.*, vol. 2594, no. 1, pp. 159–168, 2016.
[22] T. Wang, B. Yang, C. Chen, and X. Guan, "Wireless charging lane deployment in urban areas considering traffic light and regional energy supply-demand balance," in *Proc. IEEE 89th Veh. Technol. Conf.*, 2019, pp. 1–5.
[23] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" 2018, *arXiv:1810.00826*.

[24] S.-B. Cools, C. Gershenson, and B. D'Hooghe, *Self-Organizing Traffic Lights: A Realistic Simulation*. Berlin, Germany: Springer, 2013, pp. 45–55.
[25] Y. K. Chin, W. Y. Kow, W. L. Khong, M. K. Tan, and K. T. K. Teo, "Q-learning traffic signal optimization within multiple intersections traffic network," in *Proc. 6th UKSim/AMSS Eur. Symp. Comput. Model. Simul.*, 2012, pp. 343–348.
[26] I. Arel, C. Liu, T. Urbanik, and A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," *IET Intell. Transport Syst.*, vol. 4, no. 2, pp. 128–135, 2010.
[27] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018.

Xin Guo received the BS degree in mathematics and applied mathematics from Zhejiang University, Hangzhou, China, in 2015. He is currently working toward the PhD degree with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China and also the State Key Laboratory of CAD & CG, Hangzhou, China. He is currently a research intern with Alibaba DAMO Academy. His research interests include computer vision and machine learning.

Zhengxu Yu received the PhD degree from Zhejiang University, Hangzhou, China. He is currently an algorithm engineer with Alibaba DAMO Academy. His research interests include large-scale machine learning and computer vision.

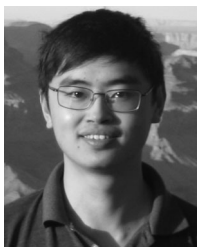
Pengfei Wang received the PhD degree from the Chinese Academy of Sciences. He is currently an algorithm engineer II with Alibaba DAMO Academy. His research interests include spatio-temporal data mining and visual intelligence.



Zhongming Jin received the PhD degree from Zhejiang University, Hangzhou, China, in Mar. 2015. He is currently a staff algorithm engineer with Alibaba DAMO Academy. Previously, he was a researcher with Baidu Research. His research interests include large-scale machine learning and computer vision.

Jianqiang Huang is currently a director with Alibaba DAMO Academy. He received the second prize in National Science and Technology Progress Award in 2010. His research interests include visual intelligence in the city brain project of Alibaba.

Deng Cai (Senior Member, IEEE) received the PhD degree in computer science from the University of Illinois at Urbana Champaign, Champaign, IL, in 2009. He is currently a professor with the State Key Lab of CAD&CG, College of Computer Science at Zhejiang University, China. His research interests include machine learning, data mining, and information retrieval.



Xiaofei He (Senior Member, IEEE) received the BS degree in computer science from Zhejiang University, Hangzhou, China, in 2000 and the PhD degree in computer science from the University of Chicago, Chicago, Illinois, in 2005. He is currently a professor with the State Key Lab of CADCG at Zhejiang University, China. Prior to joining Zhejiang University, he was a research scientist with Yahoo! Research Labs, Burbank, CA. His research interests include machine learning, information retrieval, and computer vision.

Xian-Sheng Hua (Fellow, IEEE) received the BS and PhD degrees in applied mathematics from Peking University, Beijing, China, in 1996 and 2001, respectively. In 2001, he joined Microsoft Research Asia as a researcher and has been a senior researcher of Microsoft Research Redmond since 2013. He became a researcher and the senior director of the Alibaba Group in 2015. He has authored or coauthored more than 250 research papers and has filed more than 90 patents. His research interests include multimedia search, advertising, understanding, mining, pattern recognition, and machine learning. He was honored as one of the recipients of MIT35. He served as a program co-chair for IEEE ICME 2013, ACM Multimedia 2012, IEEE ICME 2012, and on Technical Directions Board of IEEE Signal Processing Society. He is an ACM Distinguished Scientist.