scientific reports

Check for updates

OPEN

A scalable approach to optimize traffic signal control with federated reinforcement learning

Jingjing Bao[1], Celimuge Wu[1]✉, Yangfei Lin[1], Lei Zhong[2], Xianfu Chen[3] & Rui Yin[4]

Intelligent Transportation has seen significant advancements with Deep Learning and the Internet of Things, making Traffic Signal Control (TSC) research crucial for reducing congestion, travel time, emissions, and energy consumption. Reinforcement Learning (RL) has emerged as the primary method for TSC, but centralized learning poses communication and computing challenges, while distributed learning struggles to adapt across intersections. This paper presents a novel approach using Federated Learning (FL)-based RL for TSC. FL integrates knowledge from local agents into a global model, overcoming intersection variations with a unified agent state structure. To endow the model with the capacity to globally represent the TSC task while preserving the distinctive feature information inherent to each intersection, a segment of the RL neural network is aggregated to the cloud, and the remaining layers undergo fine-tuning upon convergence of the model training process. Extensive experiments demonstrate reduced queuing and waiting times globally, and the successful scalability of the proposed model is validated on a real-world traffic network in Monaco, showing its potential for new intersections.

The continual growth of the automobile industry and transportation infrastructure undoubtedly improves daily travel convenience. However, traffic congestion remains a persistent issue that poses significant challenges to individuals, economies, and the environment in various countries[1]. For instance, a recent survey by the real-time traffic data provider INRIX revealed alarming figures for London in 2022: a staggering per capita traffic delay time of 151 h, a cost of 869 US dollars per capita in delays, 546 US dollars in fuel expenses, and an alarming frequency of generating 1 metric ton of carbon emissions every 2.53 days[2]. Such congestion is caused, in part, by the growing demand for transportation that surpasses the available road capacity, necessitating improvements through infrastructure expansion or vehicle reduction[3]. Additionally, suboptimal traffic management strategies may lead to underutilization of roads and facilities, further exacerbating chaos and congestion. To address these challenges effectively, innovative and efficient traffic management solutions are crucial.

Traffic management often involves traffic control, and commonly used traffic control strategies include road-based and vehicle-based control strategies[4]. As a widely used road-based control method, Traffic Signal (TS) provides the most basic order and safety guarantee for road traffic. Advancements in technologies like Internet of Vehicles, Cloud Computing, and Deep Learning (DL) open up new possibilities for enhancing traffic conditions through intelligent Traffic Signal Control (TSC)[5]. Despite these opportunities, many cities still rely on traditional TSC schemes, such as fixed-time and heuristic control methods[6]. Fixed-time method uses historical traffic statistics to determine changing sequence of signals and the duration of every signal, while heuristic control methods lay loop sensors and use the real-time data of the road to make signal changing[7]. Although simple to implement, these methods often fail to cope with the complexities of increasing traffic flow and uncertain traffic conditions on a large scale, thereby yielding unsatisfactory results. In addressing intricate traffic scenarios, numerous optimization-based approaches are advanced with the objective of achieving optimal metrics such as travel delay, throughput, and average travel time[8]. Techniques such as Genetic Algorithms and Swarm Optimization are employed to discern the optimal solution for TSC[9,10]. Furthermore, max pressure, as an optimal control idea to maximize global throughput, is embraced by numerous research as a means to enhance traffic signal management[11]. While these methodologies demonstrate commendable outcomes under certain conditions, they often entail substantial computational expenses and extended duration to ascertain the

[1]Department of Computer and Network Engineering, The University of Electro-Communications, Tokyo 182-8585, Japan. [2]Connected Advanced Development Division, Toyota Motor Corporation JP, Tokyo 471-8571, Japan. [3]VTT Technical Research Centre of Finland, 90571 Oulu, Finland. [4]School of Information and Electrical Engineering, Hangzhou City University, Hangzhou 310015, China. ✉email: celimuge@uec.ac.jp

globally optimal solutions. Moreover, their efficacy in managing progressively intricate traffic dynamics remains a challenge.

Reinforcement Learning (RL), a powerful Machine Learning (ML) technique, shows promise in addressing TSC challenges[12]. While distributed RL methods train local decision models for each intersection independently, they lack global optimality and face difficulties in transferring knowledge to other intersections[13]. On the other hand, centralized approaches require data collection and communication infrastructure that may not scale well, especially with a growing number of intersections[14]. As a solution to these challenges, Federated Learning (FL), an integration model of DL, is emerging, enabling the integration of local knowledge without transmitting sensitive data[15].

Despite numerous studies integrating FL and RL, the existing research is limited to scenarios where all intersections in the traffic network share the same structure during each training instance, i.e., every intersection in the trained traffic network has an identical configuration[16,17]. While these approaches allow for developing a global model, their applicability is restricted to specific types of intersections. Scholars have attempted a unified representation of agent states to address this limitation, specifically exploring intersection-agnostic state representations for heterogeneous intersections[18]. However, on the one hand, the accumulation of too many factors hinders a deep network's ability to comprehend the environmental state more effectively[7]. On the other hand, challenges persist in integrating models resulting from variations in action spaces. Therefore, while existing efforts enable the integration of knowledge from locally trained models across multiple intersections, they face challenges in extending applicability to intricate intersection structures and fail to generalize effectively to other intersections with significant deviations from the training dataset.

To overcome these challenges primarily due to diverse intersection types, varying lane configurations, and dynamic phase settings that lead to differences in local model inputs and outputs, we propose a novel TSC scheme based on FL and RL to combine models effectively. By training RL models on limited intersection signal agents and integrating the feature networks learned by all agents through FL, we develop a model that can be seamlessly adapted to other intersections with minor parameter adjustments. The key contributions of this paper are as follows:

- We propose an RL method based on FL, effectively integrating the parameters of local RL models into a cloud federated model to solve the TSC problem more efficiently.
- To ensure the applicability of the integrated FL model to diverse intersection structures and phase groups, we introduce a unified state representation and an action selection method that preserves differences. Through partial aggregation and partial fine-tuning, the RL decision model of TS agent is equipped with the global feature extraction layers that contain extensive representation and the local feature extraction layers that learn local specificity.
- The effectiveness and scalability of our proposed method are demonstrated through extensive simulations on the Cologne from Germany and Monaco traffic network.

The organizational structure of this article is as follows. We present an overview of the relevant research in Related work. Problem formulation describes the problem definition and Methods details the proposed methods from three aspects: RL agent and local training, FL aggregation of RL model, and fine-tuning of RL model. Experiments and results presents simulation results that empirically demonstrate the efficacy of the proposed method. Finally, the conclusion and future research direction are discussed in Conclusion.

Related work

Traditional TSC methods, including fixed-time TSC, actuating TSC, and adaptive TSC, rely on historical data, expert knowledge, and pre-set assumptions. However, their lack of adaptability leads to inefficient signal phases, particularly in high-traffic areas and complex urban environments. Researchers explore search-based methods[19] and model optimization, such as tree search[20], genetic algorithms[21], and swarm optimization[22], to address these challenges. Prediction-based TSC methods use road state models learned by dynamic Bayesian networks to predict future traffic conditions and adjust signal lights accordingly[23]. Additionally, techniques like Dynamic Programming[24] and Fuzzy Logic[25] are applied to TSC, but as urban areas grow more complex, there is a pressing need for advanced and adaptive TSC solutions to meet modern traffic challenges.

ML approaches, including RL and Meta-learning, are introduced to TSC research. RL has been widely used in many fields such as Autopilot[26], Natural Language Processing[27] and Robot Control[28], and has achieved satisfactory results. Meanwhile, RL is adopted to enhance adaptive TSC approaches due to its data-driven nature. RL's strength lies in its ability to learn decision policies through iterative experimentation, leading to actions based on environmental feedback rather than relying on predefined rules[29–31]. RL effectively addresses the complex sequential decision problem posed by TSC and outperforms conventional methods. Early RL applications in TSC focus on single intersections, optimizing traffic signal strategies through SARSA-based self-learning control strategies[32]. Researchers also explore RL methods based on connectivity trees and phase-sensitive RL approaches to enhance system efficiency and signal light flexibility[33].

While conventional RL methods can handle simple learning tasks, their scalability and optimality are limited when dealing with complex environments and continuous state and action spaces. The integration of DL improves agents' ability to perceive complex environmental states and solve intricate problems. The Deep Q-Network (DQN) algorithm is employed to control individual intersections, utilizing discrete high-dimensional state representation methods[34]. Researchers also redefine rewards and incorporate phase gates in RL approaches, making them more applicable in real-world traffic scenarios[35]. Vision-based fully autonomous DRL agents are

developed to perceive and respond to complex and dynamic traffic environments using real-time RGB camera stream data from intersections[36].

As the number of vehicles increases, the successful application of DRL at individual intersections becomes insufficient to meet practical demands. Scholars have started exploring scenarios involving multiple intersections, aiming to extend DRL's applicability from single intersections to multi-intersection environments. Various approaches, such as multi-agent Advantage Actor-Critic algorithms and multi-layer stacked graph convolutional neural networks, are proposed to stabilize the learning process, enhance observability, and foster collaboration among intersections[37–39]. Despite its advantages, RL faces challenges in solving TSC problems. Centralized learning models can achieve optimal global results but require excessive communication and computational resources[40]. Distributed learning solutions allocate agents to learn local models, hindering generalization across different intersections[18]. This paper seeks to establish a robust model capable of aggregating knowledge from intelligent agents at various intersections with lower communication and computational costs. The model should be adaptable to different intersections with varying structures and traffic conditions.

Problem formulation

The paper addresses the TSC problem at multiple intersections, where each traffic light in all directions of an intersection is treated as an intelligent agent. The goal is to enable each TS agent within the road network to make real-time decisions and provide the best TS scheme to reduce traffic congestion. There may be variations in the number of intersections, the number of forks (directions) at each intersection (K), and the number of lanes at a particular intersection (M). The problem is modeled using RL, and the environment in which the agent interacts is crucial for formulating the TSC problem.

The environment description includes various aspects. Take Fig. 1 as an example to illustrate below:

- *Incoming lane:* Each intersection has an incoming lane, which is the lane for vehicles entering the intersection. For a specific intersection $g \in G$ (where G is the set of all intersections), it has K fork roads, and the number of lanes for each fork is $\{M_1, M_2, ..., M_K\}$. So, the incoming lane of the kth fork is recorded as $L = \{l_1, l_2, ..., l_{M_k}\}$.
- *Movement restriction:* Due to differences in lane settings and traffic flow at each intersection, vehicles allowed to move from the incoming lane to the intersection may vary. For instance, some intersections may not permit left turns in right-hand traffic rules.
- *Phase:* Phases represent the utilization of several lights as a TS at intersections to direct vehicles through the intersection safely. Typically, only non-conflicting driving directions are allowed to pass at any moment. The number of phases may vary depending on the intersection, and certain intersections with low traffic flow may have unique phase settings that allow for simultaneous straight-going and left-turning movements. The right side of Fig. 1 illustrates the possible phases of the intersection shown on the left side.
- *Traffic flow:* Traffic flow varies based on the degree of congestion and may differ at each intersection in terms of time and density. Agents need to make optimal decisions under different traffic flow conditions.

Methods

With reference to the framework in Fig. 2, the proposed method consists of three stages: local training of RL agent, FL aggregation and fine-tuning of the local model. First, RL is used to learn an optimal TSC policy for each intersection with local data. Next, to handle the scalability issue, the approach adopts FL, where RL agents at each intersection act as clients and send partial model updates to the server, and a central server coordinates the training process which aggregates the updates and sends back the updated global model. Finally, after the FL process converges, the remaining parameters are fine-tuned according to the local characteristics of each agent to ensure good performance at new intersections. This approach leverages the collective knowledge of all agents to train more effective global expression and knowledge base applicable to all signals. In parallel, it attempts to provide each agent with a solution to diversity and particularities.

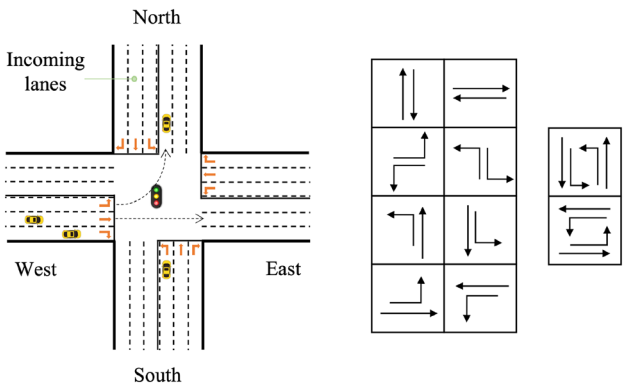


Figure 1. Illustration of intersection and phases.

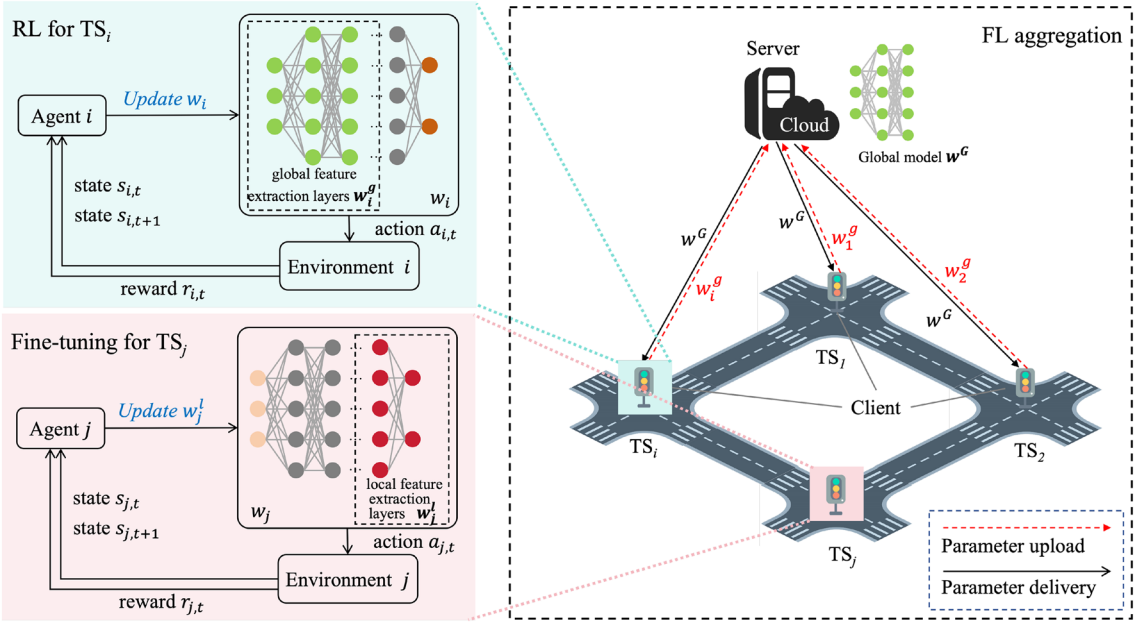


Figure 2. Framework for federated reinforcement learning of TSC.

Local training of reinforcement learning agent

RL is an ML paradigm designed to tackle sequential decision-making problems. At its core is the Markov Decision Process (MDP), which outlines the fundamental elements governing the interaction between an agent and its environment. The TSC problem is a discrete-time RL task, and its MDP can be represented by the tuple $\{S, A, P, R, \gamma\}$. Here, S signifies the state space, A is the action space, R represents the reward generated after the action affects the environment, P is the state transition probability, and γ is the discount factor that balances immediate reward and future rewards. In the TSC task, the goal of RL is to find an optimal control strategy π^*, optimizing the overall performance of vehicles passing through the intersection. The specific agent design is introduced below.

State representation To achieve global knowledge integration using FL, a critical aspect is to design a unified representation method for the local RL models learned at intersections of different road structures. For this purpose, an intersection-independent state representation method is established, allowing different action spaces for each agent.

The state representation includes various observation features of the intersection, such as queue length, number of vehicles, position and speed of vehicles, delay time, waiting time, and the current signal phase[41]. This information is collected from various sensors like cameras and radars placed at intersections. The state is represented as a multidimensional matrix:

$$s = \begin{bmatrix} s_{11}^1 & \cdots & s_{1m}^1 & s_{11}^2 & \cdots & s_{1m}^2 & \cdots & s_{11}^k & \cdots & s_{1m}^k \\ s_{21}^1 & \cdots & s_{2m}^1 & s_{21}^2 & \cdots & s_{2m}^2 & \cdots & s_{21}^k & \cdots & s_{2m}^k \\ & & & & \vdots & & & & & \\ s_{n1}^1 & \cdots & s_{nm}^1 & s_{n1}^2 & \cdots & s_{nm}^2 & \cdots & s_{n1}^k & \cdots & s_{nm}^k \end{bmatrix}, \tag{1}$$

where s_{nm}^k represents the value of the nth observation feature of the mth incoming lane of the kth fork road. In cases where the observed features, number of lanes, or number of forks at the current intersection are smaller than the maximum values, the matrix is filled with 0. This design ensures a uniform representation of the state input while preserving detailed structural information of the intersection, facilitating the model's ability to learn the mapping from intersection state to TS decisions.

Action selection In RL, action refers to the choices and movements made by the agent to interact with the environment. In the TSC problem, actions are typically selected based on two dimensions: time and phrase. Time refers to the decision-making interval, denoted as ΔT seconds, allowing the model to flexibly change the phase while ensuring vehicles can safely exit the intersection within one cycle. Phase selection involves each agent choosing one phase from its valid phase set. The number and types of phases vary across intersections based on their road structures and traffic characteristics. If an agent $g \in G$ has p phases, its action set is $A = \{a_1, a_2, \ldots, a_p\}$, where each element corresponds to the selection of one phase.

Reward function The reward function plays a crucial role in training RL models. The reward in this paper is defined as the weighted sum of the average queue length of halted vehicles (H) and the average waiting time of the first vehicle in each lane (T) at the intersection. The reward for agent $g \in G$ is:

$$r = \sum_{l_m \in L} (H_{l_m} + \sigma T_{l_m}), \tag{2}$$

where H_{l_m} and T_{l_m} are the queue length of halting vehicles and waiting time of the first vehicle on l_m lane at g intersection, respectively. L is the incoming lane set of intersection agent g. And σ is a weight parameter that discounts the waiting time, and its value ranges from 0 to 1. As the vehicles in the experiments have identical properties, the queue length of halted vehicles is replaced by the number of halted vehicles in the queue.

Reinforcement training algorithm The DQN[42] is employed in the proposed method for its simplicity and effectiveness. DQN is a type of model-free RL, and the optimal policy $\pi^*$is learned through the Q-function. The Q-function is defined as the expected future reward for taking action at in state s_t, estimated by iterative Bellman update:

$$Q(s_{t+1}, a_{t+1}) = Q(s_t, a_t) + \alpha \left(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right), \tag{3}$$

where $Q(s_t, a_t)$ is Q-function in time step t, α is the learning rate, γ is the discount factor, t is the current time step, $t + 1$ is the next time step, and s_t, a_t and r_t are the state, action and reward of the agent in time step t respectively.

In DQN, a deep neural network is used to approximate the Q-function, denoted as $Q(s, a; w)$, where w are the network weights. To stabilize the learning process, DQN introduces a target network, the weights of which, w_t^-, are a slowly updated version of the main network weights. This target network is used to compute the target Q-values, in the Bellman update rule. Therefore, the loss function used to train the neural network is:

$$L(w) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim \text{replay buffer}} \left[\left(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; w^-) - Q(s_t, a_t; w) \right)^2 \right], \tag{4}$$

where w_t^- is parameter of target network. An experience replay buffer is used to store past transitions (state, action, reward, next state) to break the correlation between consecutive data and improve the stability of learning. During each training iteration, a batch of data is randomly sampled from this buffer for learning.

Federated learning aggregation

To address scalability issues and avoid transmitting original training data to a centralized server, FL is adopted, which allows model parameters to be transferred instead. FL achieves knowledge integration by aggregating model parameters learned locally on individual data.

Although the FL can improve the generalization ability of the model, various heterogeneous intersections exist in reality. The most obvious intuition is that due to different road structures and requirements, the phase combinations of TS at each intersection are quite different, so it is difficult to unify the output space of the RL model. On the other hand, each area or intersection has certain unique characteristics, which may contain important information about traffic management, which plays an important role in local optimization. These realities put limitations on the model that directly integrates agents at each intersection, and the trained global model may not be able to cope with the situation of new intersections. To this end, the TS agent's network model is divided into two parts: the global feature extraction layers and the local feature extraction layers. In the federated learning process, only the global feature extraction layers w^g are integrated into the cloud so that it has a generalized representation at the global level, while the local feature extraction layers w^l, as a part of the network that retains the unique knowledge of each agent, does not participate in the integration.

As presented in the graphical depiction of Fig. 2, the process involves clients actively sending their global feature extraction layers' parameters (w_i^g) to the cloud server after completing a certain number of local RL training, which then conducts aggregation to obtain global feature parameters (w^G). Afterward, the cloud server delivers the integrated model parameters to each TS client participating in the training. At this time, each TS agent only needs to update the received model parameters to global feature extraction layers. A one-time federation process is performed at fixed learning intervals (RL training interval) for each round of learning (FL aggregation round). The aggregation algorithm used in this paper is AvgFed[43], a weighted average fusion method based on the amount of data owned by the local model:

$$w^G = aggregate(w_1^g, w_2^g, ..., w_N^g) = \frac{1}{N} \sum_{i \in N} w_i^g, \tag{5}$$

where N is the number of TS agents in the current scenario and w_i^g is RL model parameters of global feature extraction layers of TS agent g_i. w^G is the neural network parameters of the global federated model. Assuming the observation data of each agent in the TSC environment is uniformly distributed, the aggregation is defined accordingly.

Fine-tuning of the local feature extraction layers

In practical applications of TSC, training a control model for each intersection is unrealistic and not energy-friendly. Therefore, we hope to obtain a model with extensive knowledge and generalization capabilities. To achieve this goal, FL aggregation is first utilized to obtain a global model that learns common features regarding TSC from a wide range of datasets. During the training process, the local feature extraction layers remain local and do not participate in federated aggregation until the model converges. A critical consideration is, when

encountering new TSs, local model fine-tuning is performed. This fine-tuning process is to train the neurons of the local feature extraction layers of the new TS agent to adapt to the specific phase settings and other traffic requirements of the TS, thereby achieving better performance in a short time. During this process, the global feature layers remain unchanged, namely, no backpropagation is used to update the global feature layers, which means that the common features learned from the initial broad dataset are retained. During the fine-tuning process, the loss function of TS as an RL agent to update network parameters is as follows:

$$L(w^l) = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})}\left[\left(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; w^{l,-}) - Q(s_t, a_t; w^l)\right)^2\right], \tag{6}$$

where w^l is the parameters of local feature extraction layers and $w^{l,-}$ is the local feature extraction layers' parameters of the target network.

This approach aims to enhance the adaptability and efficiency of the system by modifying the new local feature extraction layers to rapidly apply the global model to specific contexts of the new agent. This facilitates the swift adaptation of the global model to specific environments, enabling it to accommodate new TSs with diverse features while incurring low computational costs and training time requirements.

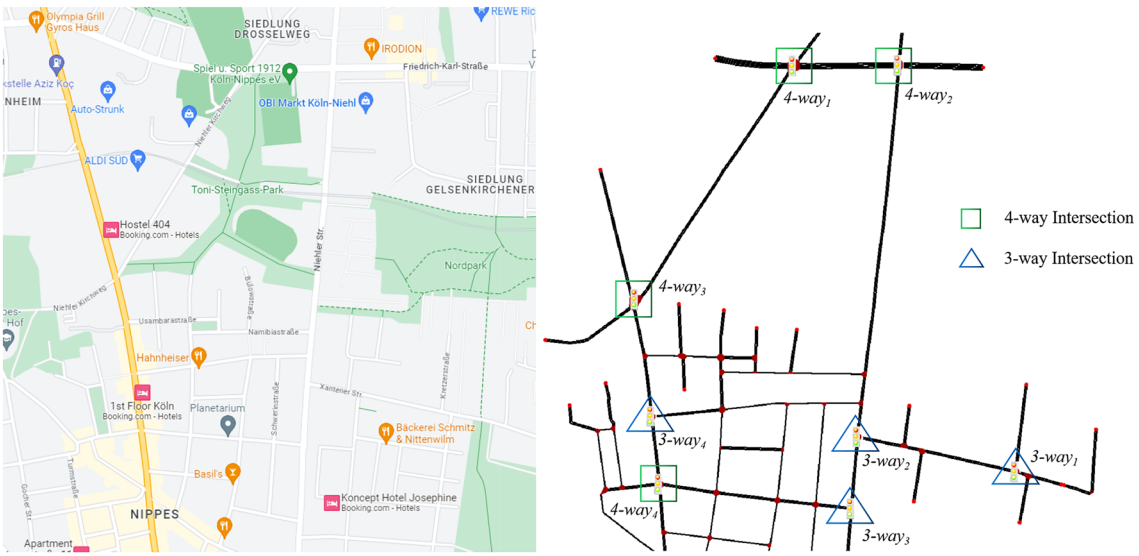
Experiments and results

Experimental setting

The experiments are conducted using the SUMO (Simulation of Urban Mobility) simulator, known for its realistic environment and road behavior simulation capabilities. A real road network Fig. 3a in Cologne, Germany, comprising 8 intersections with diverse structures Fig. 3b, is used for the simulations.

The neural network of DQN consists of an input layer, three fully connected feature extraction layers, and an output layer. The numbers of neurons in the hidden layers are 64, 128, and 256, respectively. The output layer parameters are left on the local agent side as the local feature extraction layer and do not participate in aggregation. FL round is set as 20 episodes, where local agents upload their parameters after every 20 training episodes for aggregation and distribution. Other experimental parameters are shown in Table 1.

Four types of models are compared in the experiments:



(a) Map of selected area in Cologne. **(b)** Network of Cologne with 8 traffic signals in SUMO.

Figure 3. Cologne road network with 8 intersections' traffic signals.

Parameters	Value
Learning rate	0.0001
ε-*greedy* action selection	0.9
Reward discount of RL γ	0.9
Target network update frequency	30 episodes
Memory capacity	1000
Discount weight of waiting time in reward σ	0.1

Table 1. Parameters of experiment.

- *IDQN:* The Independent DQN models without FL are deployed at each intersection, with each agent training its own model independently based on limited local observations.
- *IDQN_tuned:* The IDQN model is fine-tuned to improve its performance in the test, ensuring a fair comparison with the proposed method.
- *Fed_trained:* In the FL process, although the proposed method does not aggregate output layer parameters, each agent holds its own output layer locally. This model serves as a comparison for experiments.
- *Proposed:* The federated global model is deployed at each intersection, and the output layer parameters of each agent are randomly initialized and fine-tuned without the feature layer participating in the fine-tuning.

Results

Training process results

During the training process, the average reward values for each intersection are recorded. As shown in Fig. 4, compared to the fixed-time method with a reward value of −1319, the IDQN and the proposed method converged to −575.17 and −562.28, respectively, indicating the superiority of the DL approach over the fixed-time method. The federated model shows better final convergence, improving by 2.29%. This is likely because the federated model effectively utilizes information from individual intersections, allowing it to adapt better to various traffic conditions and maintain more stable performance.

The trained models are tested on the original traffic network using metrics such as the halting number of vehicles at each intersection, the average waiting time of the vehicles at the head of the queue, and the time delay of all vehicles. The results of the four models are plotted in Fig. 5, and the federated model shows significant improvements in all three indicators compared to the independently trained IDQN model. The proposed model also outperforms the Fed_trained model, demonstrating its ability to absorb global knowledge and achieve better results after a few rounds of fine-tuning.

Evaluation in different traffic conditions

To verify the adaptability of the proposed model in different traffic conditions, the density of traffic flow is set as a variable for experiments. The number of vehicles entering the road network per hour is changed to 1200, 1800, and 2400 vehicles per hour. The evaluation indicators include the average halting number of vehicles, the average waiting time of the first vehicle in each lane, and the average cumulative waiting time in all intersections. And

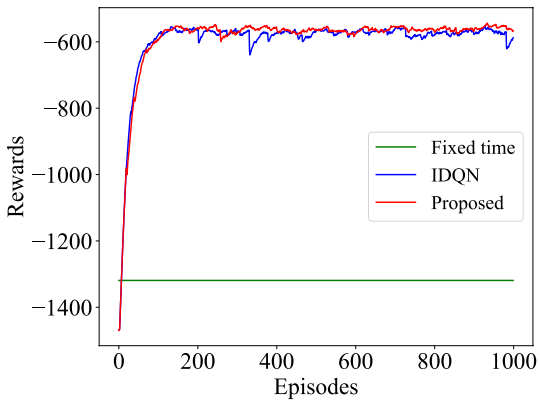


Figure 4. Variation of rewards during training process.

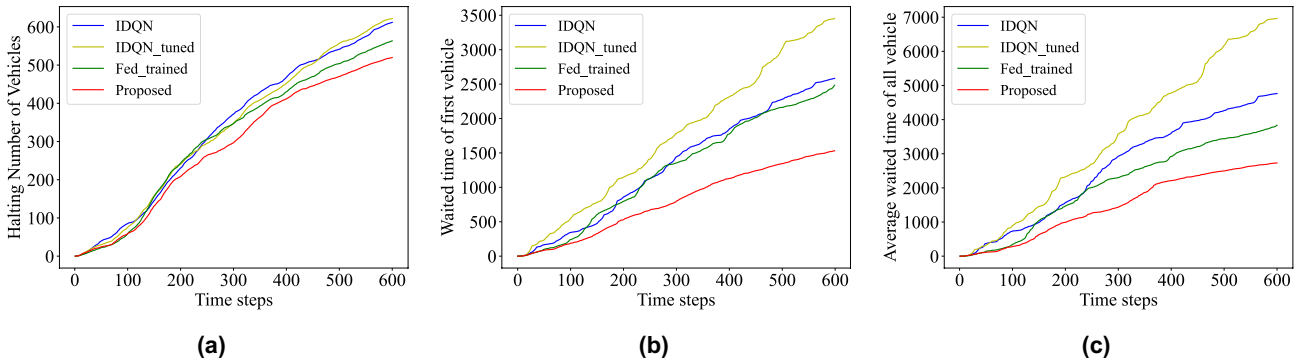


Figure 5. The performance of the proposed method in training data on: (**a**) halting number of vehicles; (**b**) waiting time of first vehicle in every lane; (**c**) average cumulative waiting time of all intersections.

the results are the current evaluation value at each time step. The experimental results show that the proposed method consistently outperformed the comparison methods in traffic scenes with different densities. As shown in Fig. 6, when the traffic flow is 1200 veh/h, the proposed method reduces the halting vehicles, the waiting time of the first vehicle, and the waiting time of all vehicles by an average of 39.95%, 55.65%, and 64.48%, respectively, compared to the other models.

Experiments of model transplantation

The model trained on 8 TS agents in Cologne is tested for scalability by deploying it on another real-world traffic network in Monaco, which includes 8 four-way intersections and 6 three-way intersections. For the IDQN method, the two groups of models with the best performance are selected for comparison among all combinations and deployed at the intersection of the same structure of the Monaco network. The results are shown in Fig. 7, which demonstrates that the proposed method significantly outperforms the combination of models trained by the IDQN on the Monaco network. As demonstrated in Fig. 7a, even after fine-tuning, the IDQN model fails to achieve similar results. Given that the output layer of the IDQN is an outcome of post-training parameterization, the initial step of fine-tuning yields enhanced performance in comparison to the federated model, which employs random initialization for the output layer. However, despite the initial advantages, the IDQN training ultimately failed due to the fact that only limited features of inputs are learned in the feature layer of the locally trained model and new knowledge cannot be further learned by fine-tuning the output layer independently. On the contrary, the proposed method converged to a better level. In Fig. 7b,c, the testing result of the halting number of vehicles and average waiting time of all vehicles are given. The proposed method maintains fewer vehicles waiting at the intersection, while the comparison method increases significantly. The results indicate that the locally trained IDQN models lack the adaptability required to handle new environments effectively. In contrast,

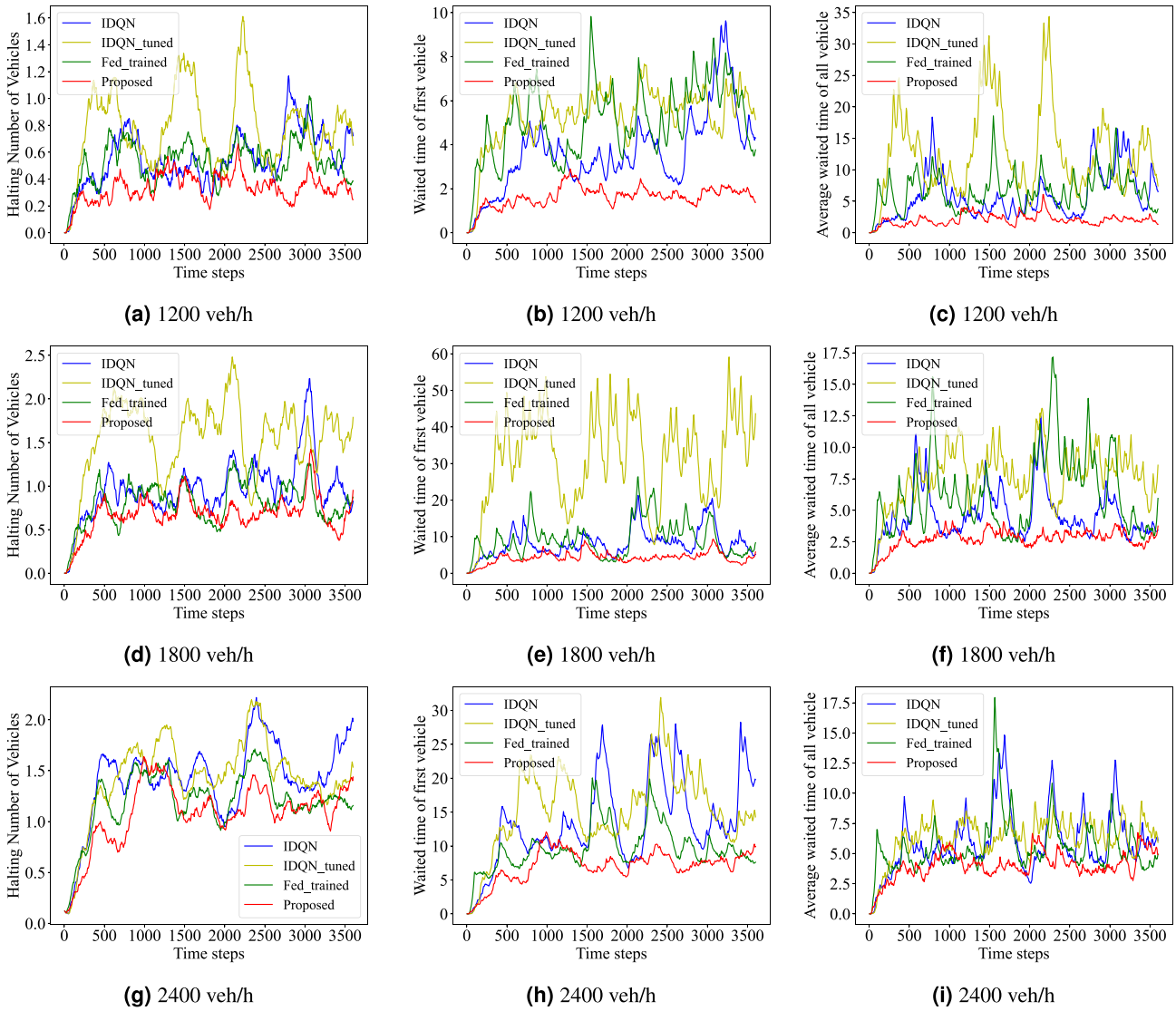


Figure 6. The performance with different density of traffic flow on: halting number of vehicles; waiting time of first vehicle in every lane; average waiting time of all intersections.

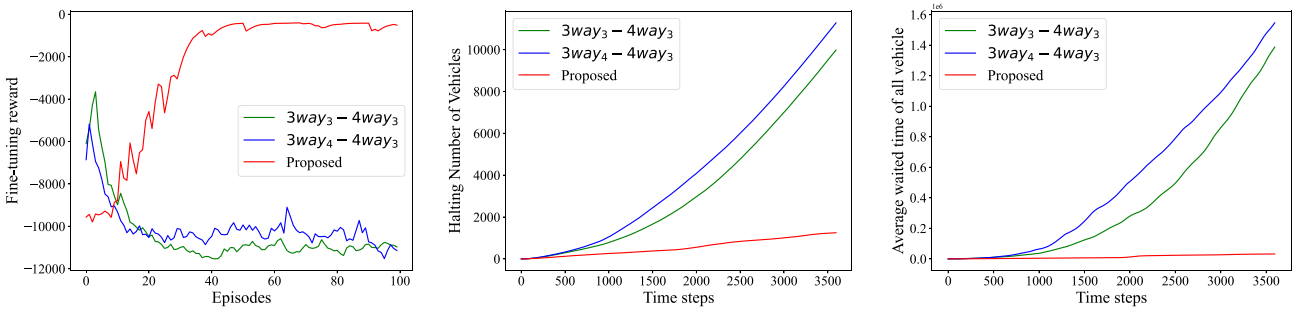


Figure 7. The performance of the proposed method in new road network: (**a**) fine-tuning reward (**b**) halting number of vehicles; (**c**) average cumulative waiting time of all intersections.

the proposed method, integrated using FL and fine-tuned, demonstrates superior performance when applied to different intersections with varying road structures and traffic conditions.

Overall, the experiments validate the effectiveness and adaptability of the proposed model in various traffic conditions and its ability to outperform locally trained models when applied to new intersections. The fine-tuning process further enhances the model's ability to adapt to new environments, leading to improved decision-making performance.

Conclusion

In this paper, we present a novel approach to TSC by integrating RL with FL. The proposed method effectively addresses the scalability challenge of RL by adopting FL, which unifies the state space and integrates feature extraction layer parameters of neural networks. The network of RL model is divided into two parts: the global feature extraction layers and the local feature extraction layers. The former is integrated into a network with global knowledge through FL and sent to the cloud, while the latter learns local unique properties through fine-tuning on the client. Such a learning model better adapts to the output differences and environmental variations among different agents. The evaluation on a real-world traffic network demonstrates the superiority of the proposed approach over the Independent DQN methods in terms of traffic flow efficiency and travel time. Specifically, the convergence performance during training is improved by 2.29%; halting number of vehicles, waiting time of first vehicle in every lane and average cumulative waiting time are improved by an average of 39.95%, 55.65% and 64.48% respectively. In future research, exploring various aggregation algorithms, such as client selection or weight assignment based on performance or reliability, can further optimize model performance. Additionally, incorporating emerging aggregation algorithms based on meta-learning may contribute to further enhancing the performance of FL in tackling TSC problems. This approach represents a promising step towards efficient and adaptive TSC, with potential applications in more complex and diverse urban traffic scenarios, as it has demonstrated superior performance compared to traditional methods.

Data availability

The datasets used, generated and analyzed during this study are available from the corresponding author on reasonable request.

Received: 9 August 2023; Accepted: 27 October 2023
Published online: 06 November 2023

References

1. Haydari, A. & Yılmaz, Y. Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Trans. Intell. Transp. Syst.* **23**, 11–32. https://doi.org/10.1109/TITS.2020.3008612 (2020).
2. Inrix 2022 global traffic scorecard. Tech. Rep., INRIX (2023). https://inrix.com/scorecard/.
3. Rasheed, F., Yau, K.-L.A., Noor, R. M., Wu, C. & Low, Y.-C. Deep reinforcement learning for traffic signal control: A review. *IEEE Access* **8**, 208016–208044. https://doi.org/10.1109/ACCESS.2020.3034141 (2020).
4. Siri, S., Pasquale, C., Sacone, S. & Ferrara, A. Freeway traffic control: A survey. *Automatica* **130**, 109655. https://doi.org/10.1016/j.automatica.2021.109655 (2021).
5. Wei, H., Zheng, G., Gayah, V. & Li, Z. Recent advances in reinforcement learning for traffic signal control: A survey of models and evaluation. *ACM SIGKDD Explor. Newsl.* **22**, 12–18. https://doi.org/10.1145/3447556.3447565 (2021).
6. Eom, M. & Kim, B.-I. The traffic signal control problem for intersections: A review. *Eur. Transp. Res. Rev.* **12**, 1–20. https://doi.org/10.1186/s12544-020-00440-8 (2020).
7. Noaeen, M. *et al.* Reinforcement learning in urban network traffic signal control: A systematic literature review. *Expert Syst. Appl.*https://doi.org/10.1016/j.eswa.2022.116830 *(2022).*
8. Kouvelas, A., Aboudolas, K., Kosmatopoulos, E. B. & Papageorgiou, M. Adaptive performance optimization for large-scale traffic control systems. *IEEE Trans. Intell. Transp. Syst.* **12**, 1434–1445. https://doi.org/10.1109/TITS.2011.2159002 (2011).
9. Ceylan, H. & Bell, M. G. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. *Transp. Res. Part B Methodol.* **38**, 329–342. https://doi.org/10.1016/S0191-2615(03)00015-8 (2004).
10. Shaikh, P. W., El-Abd, M., Khanafer, M. & Gao, K. A review on swarm intelligence and evolutionary algorithms for solving the traffic signal control problem. *IEEE Trans. Intell. Transp. Syst.* **23**, 48–63. https://doi.org/10.1109/TITS.2020.3014296 (2020).

11. Varaiya, P. The max-pressure controller for arbitrary networks of signalized intersections. *Adv. Dyn. Netw. Model. Complex Transp. Syst.*https://doi.org/10.1007/978-1-4614-6243-9_2 *(2013).*
12. Guo, Q., Li, L. & Ban, X. J. Urban traffic signal control with connected and automated vehicles: A survey. *Transp. Res. Part C Emerg. Technol.* **101**, 313–334. https://doi.org/10.1016/j.trc.2019.01.026 (2019).
13. Wang, X., Ke, L., Qiao, Z. & Chai, X. Large-scale traffic signal control using a novel multiagent reinforcement learning. *IEEE Trans. Cybern.* **51**, 174–187. https://doi.org/10.1109/TCYB.2020.3015811 (2020).
14. Noaeen, M., Mohajerpoor, R., Far, B. H. & Ramezani, M. Real-time decentralized traffic signal control for congested urban networks considering queue spillbacks. *Transp. Res. Part C Emerg. Technol.* **133**, 103407. https://doi.org/10.1016/j.trc.2021.103407 (2021).
15. Kairouz, P. *et al.* Advances and open problems in federated learning. *Found. Trends Mach. Learn.* **14**, 1–210. https://doi.org/10.1561/2200000083 (2021).
16. Wang, T. *et al.* Adaptive traffic signal control using distributed marl and federated learning. In *2020 IEEE 20th International Conference on Communication Technology (ICCT)*, 1242–1248, https://doi.org/10.1109/ICCT50939.2020.9295660 (IEEE, 2020).
17. Ye, Y., Zhao, W., Wei, T., Hu, S. & Chen, M. Fedlight: Federated reinforcement learning for autonomous multi-intersection traffic signal control. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*, pp 847–852. https://doi.org/10.1109/DAC18074.2021.9586175 (IEEE, 2021).
18. Hudson, N., Oza, P., Khamfroush, H. & Chantem, T. Smart edge-enabled traffic light control: Improving reward-communication trade-offs with federated reinforcement learning. In *2022 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp 40–47, https://doi.org/10.1109/SMARTCOMP55677.2022.00021 (IEEE, 2022).
19. Gao, K., Zhang, Y., Sadollah, A. & Su, R. Optimizing urban traffic light scheduling problem using harmony search with ensemble of local search. *Appl. Soft Comput.* **48**, 359–372. https://doi.org/10.1016/j.asoc.2016.07.029 (2016).
20. Cheng, Y., Hu, X., Tang, Q., Qi, H. & Yang, H. Monte carlo tree search-based mixed traffic flow control algorithm for arterial intersections. *Transp. Res. Rec.* **2674**, 167–178. https://doi.org/10.1177/0361198120919746 (2020).
21. Putha, R., Quadrifoglio, L. & Zechman, E. Comparing ant colony optimization and genetic algorithm approaches for solving traffic signal coordination under oversaturation conditions. *Comput. Aided Civ. Infrastruct. Eng.* **27**, 14–28. https://doi.org/10.1111/j.1467-8667.2010.00715.x (2012).
22. Celtek, S. A., Durdu, A. & Alı, M. E. M. Real-time traffic signal control with swarm optimization methods. *Measurement* **166**, 108206. https://doi.org/10.1016/j.measurement.2020.108206 (2020).
23. Chaudhary, S., Indu, S. & Chaudhury, S. Video-based road traffic monitoring and prediction using dynamic Bayesian networks. *IET Intel. Transp. Syst.* **12**, 169–176. https://doi.org/10.1049/iet-its.2016.0336 (2018).
24. Cai, C., Wong, C. K. & Heydecker, B. G. Adaptive traffic signal control using approximate dynamic programming. *Transp. Res. Part C Emerg. Technol.* **17**, 456–474. https://doi.org/10.1016/j.trc.2009.04.005 (2009).
25. Kumar, N., Rahman, S. S. & Dhakad, N. Fuzzy inference enabled deep reinforcement learning-based traffic light control for intelligent transportation system. *IEEE Trans. Intell. Transp. Syst.* **22**, 4919–4928. https://doi.org/10.1109/TITS.2020.2984033 (2020).
26. Yan, Z., Kreidieh, A. R., Vinitsky, E., Bayen, A. M. & Wu, C. Unified automatic control of vehicular systems with reinforcement learning. *IEEE Trans. Autom. Sci. Eng.* **20**, 789–804. https://doi.org/10.1109/TASE.2022.3168621 (2022).
27. Uc-Cetina, V., Navarro-Guerrero, N., Martin-Gonzalez, A., Weber, C. & Wermter, S. Survey on reinforcement learning for language processing. *Artif. Intell. Rev.* **56**, 1543–1575. https://doi.org/10.1007/s10462-022-10205-5 (2023).
28. Brunke, L. *et al.* Safe learning in robotics: From learning-based control to safe reinforcement learning. *Ann. Rev. Control Robot. Autonom. Syst.* **5**, 411–444. https://doi.org/10.1146/annurev-control-042920-020211 (2022).
29. Wu, T. *et al.* Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks. *IEEE Trans. Veh. Technol.* **69**, 8243–8256. https://doi.org/10.1109/TVT.2020.2997896 (2020).
30. Zhu, L., Peng, P., Lu, Z. & Tian, Y. Metavim: Meta variationally intrinsic motivated reinforcement learning for decentralized traffic signal control. *IEEE Trans. Knowl. Data Eng.*https://doi.org/10.1109/TKDE.2022.3232711 *(2023).*
31. Tomar, I., Sreedevi, I. & Pandey, N. State-of-art review of traffic light synchronization for intelligent vehicles: Current status, challenges, and emerging trends. *Electronics* **11**, 465. https://doi.org/10.3390/electronics11030465 (2022).
32. Thorpe, T. L. & Anderson, C. W. Traffic light control using sarsa with three state representations. *Technical report, Citeseer* (1996).
33. Zhao, Y., Ma, J., Shen, L. & Qian, Y. Optimizing the junction-tree-based reinforcement learning algorithm for network-wide signal coordination. *J. Adv. Transp.* **1–11**, 2020. https://doi.org/10.1155/2020/6489027 (2020).
34. Genders, W. & Razavi, S. Using a deep reinforcement learning agent for traffic signal control. arXiv:1611.01142 (2016).
35. Wei, H., Zheng, G., Yao, H. & Li, Z. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2496–2505, https://doi.org/10.1145/3219819.3220096 (2018).
36. Garg, D., Chli, M. & Vogiatzis, G. Fully-autonomous, vision-based traffic signal control: From simulation to reality. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, 454–462 (2022).
37. Chu, T., Wang, J., Codecà, L. & Li, Z. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Trans. Intell. Transp. Syst.* **21**, 1086–1095. https://doi.org/10.1109/TITS.2019.2901791 (2019).
38. Nishi, T., Otaki, K., Hayakawa, K. & Yoshimura, T. Traffic signal control based on reinforcement learning with graph convolutional neural nets. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 877–883, https://doi.org/10.1109/ITSC.2018.8569301 (IEEE, 2018).
39. Rasheed, F., Yau, K.-L.A., Noor, R. M. & Chong, Y.-W. Deep reinforcement learning for addressing disruptions in traffic light control. *Comput. Mater. Cont.*https://doi.org/10.32604/cmc.2022.022952 *(2022).*
40. Haddad, T. A., Hedjazi, D. & Aouag, S. A deep reinforcement learning-based cooperative approach for multi-intersection traffic signal control. *Eng. Appl. Artif. Intell.* **114**, 105019. https://doi.org/10.1016/j.engappai.2022.105019 (2022).
41. Qadri, S. S. S. M., Gökçe, M. A. & Öner, E. State-of-art review of traffic signal control methods: Challenges and opportunities. *Eur. Transp. Res. Rev.* **12**, 1–23. https://doi.org/10.1186/s12544-020-00439-1 (2020).
42. Mnih, V. *et al.* Playing atari with deep reinforcement learning. arXiv:1312.5602 (2013).
43. McMahan, B., Moore, E., Ramage, D., Hampson, S. & y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282 (PMLR, 2017).

Acknowledgements

This research was supported in part by JSPS Bilateral Joint Research Project No. JPJSB120231002, in part by collaborative research with ToyotaMotor Corporation, in part by the ROIS NII Open Collaborative Research 23S0601, and in part by JSPS KAKENHI grant number 21H03424.

Author contributions

The authors confirm contribution to the paper as follows: J.B. and C.W.: conceiving, conceptualisation. J.B.: methodology and conducting the experiments. R.Y. and Y.L.: data collation, data analysis and curation. X.C.:

drafting of paper content. Y.L. and L.Z.: writing, reviewing and editing. C.W.: supervision and mentoring. All authors reviewed the manuscript.

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to C.W.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.