

Article



A Traffic-Aware Federated Imitation Learning Framework for Motion Control at Unsignalized Intersections with Internet of Vehicles

Tianhao Wu * , Mingzhi Jiang , Yinhui Han, Zheng Yuan, Xinhang Li and Lin Zhang

School of Artificial Intelligence, Beijing University of Posts and Telecommunications, 10 Xitucheng Road, Haidian District, Beijing 100876, China; mingzhijiang@bupt.edu.cn (M.J.); hanyh@bupt.edu.cn (Y.H.); yuanzheng@bupt.edu.cn (Z.Y.); lixinhang@bupt.edu.cn (X.L.); zhanglin@bupt.edu.cn (L.Z.) * Correspondence: wu.tianhao@bupt.edu.cn

Abstract: The wealth of data and the enhanced computation capabilities of Internet of Vehicles (IoV) enable the optimized motion control of vehicles passing through an intersection without traffic lights. However, more intersections and demands for privacy protection pose new challenges to motion control optimization. Federated Learning (FL) can protect privacy via model interaction in IoV, but traditional FL methods hardly deal with the transportation issue. To address the aforementioned issue, this study proposes a Traffic-Aware Federated Imitation learning framework for Motion Control (TAFI-MC), consisting of Vehicle Interactors (VIs), Edge Trainers (ETs), and a Cloud Aggregator (CA). An Imitation Learning (IL) algorithm is integrated into TAFI-MC to improve motion control. Furthermore, a loss-aware experience selection strategy is explored to reduce communication overhead between ETs and VIs. The experimental results show that the proposed TAFI-MC outperforms imitated rules in the respect of collision avoidance and driving comfort, and the experience selection strategy can reduce communication overheads while ensuring convergence.

Keywords: federated learning; imitation learning; internet of vehicle; unsignalized intersection

1. Introduction

The Internet of Vehicles (IoV) provides ubiquitous connectivity in transportation scenarios, allowing massive data interaction among smart vehicles, road infrastructures, and remote computing facilities. Empowered by IoV, the issues of safety [1] and efficiency [2] can be addressed in a cooperative manner. This study focuses on the optimization of unsignalized intersection management [3], where motions of each vehicle are precisely controlled to pass through an intersection without traffic lights.

Compared to traffic light controls, unsignalized intersection management promises higher efficiency while ensuring driving safety. The process of deciding the vehicle passing sequence is known as scheduling. Two main categories of existing scheduling policies are negotiation-based [4] and reservation-based [5]. Because the actions taken by vehicles depend on real-time driving conditions, which is a typical Markov Decision Process (MDP), Reinforcement Learning (RL) is suitable to address this issue. Guan et al. [6] proposed an RL-based method for guiding a fixed number of vehicles through an intersection. Wu et al. [7] proposed a cooperative RL method to improve traffic efficiency while ensuring safety. This method decoupled the relationship between identity and driving information on vehicles. Jiang et al. [8] proposed a two-stage RL incorporating end-edge-cloud architecture to achieve global optimization among multiple homogeneous intersections. However, low sample efficiency and limited safety performance make practical applications difficult.

Compared with RL's low sample inefficiency, Imitation Learning (IL) can accelerate the training process and does not need to specify how the task should be performed. This is because the expected behaviors are embedded in expert demonstrations. IL has been



Citation: Wu, T.; Jiang, M.; Han, Y.; Yuan, Z.; Li, X.; Zhang, L. A Traffic-Aware Federated Imitation Learning Framework for Motion Control at Unsignalized Intersections with Internet of Vehicles. *Electronics* **2021**, *10*, 3050. https://doi.org/ 10.3390/electronics10243050

Academic Editors: Dong Seog Han, Kalyana C. Veluvolu and Takeo Fujii

Received: 3 November 2021 Accepted: 29 November 2021 Published: 7 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). one of the popular approaches used to train control policy for many fields for efficiency and simplicity, such as resource scheduling [9], signalized intersection control [10], and smart manufacturing [11]. Nowadays, rule-based strategies are still the mainstream of self-driving development in the industry because of their interpretability. To facilitate self-driving development, many companies established huge scenario libraries [12] and simulator [13]. Due to the existence of thresholds in various rules, there is a bottleneck in improving driving comfort. IL can mitigate the thresholds' impact during translating rules into neural network knowledge.

Although some progress has been made in learning-based motion control at intersections, three problems remain:

- Isolation: To balance motion control performance and privacy preservation, setting a local center to assist the motion control optimization is essential, which means that some privacy-sensitive data are delivered to the local center. However, for privacy requirements, data exposure to cloud nodes or other peer nodes is prohibited. This constructs data isolation among intersections;
- (2) Heterogeneity: Due to vehicles' non-uniform spatial distribution, intersections in different areas carry different traffic flows. One of the traffic flow characteristics is the flow rate difference. Due to different traffic flows at different intersections, generated experience data drives obtained the RL model to demonstrate different capabilities for motion control optimization. Therefore, conventional model parameters averaging cannot meet the performance requirements at different intersections;
- (3) Scalability: As the number of IoV-enabled unsignalized intersections grows, data generated by the vehicles increase. Because of the incurred high computation and communication budget, any learning-based algorithm with a centralized property may find it challenging to handle such data.

Communication overhead is an unavoidable topic in distributed deep learning. Many researchers try to reduce the communication overhead while ensuring model convergence. Luo et al. [14] proposed a gradient compression method, which reduced the communication overhead between the master node and the multiple compute nodes. Shi et al. [15] proposed the optimal merged gradient sparsification algorithm based on SGD to solve the high communication overhead caused by gradient sparsity in deep learning. Sattler et al. [16] proposed a robust compression framework, sparse ternary compression, aiming at the limited scope of existing schemes to reduce communication overhead between the server and clients in federated learning. Many existing studies assume that data stay with a deep learning algorithm. This paper considers a scenario in which data and algorithms are separated. Trainers need to collect data from vehicles to generate a neural network model. Inspired by [17], not all experience is equally helpful for model training. A loss-aware experience selection strategy is proposed to reduce the communication overhead by discarding low-value experiences.

In this study, Federated Learning (FL) is chosen to address the challenges mentioned above. FL is a distributed deep learning paradigm, which enables multiple clients to learn a shared model while storing all the training data on clients. Many researchers transfer FL into many areas, such as communication network slicing [18], traffic classification [19], and crowd computing [20] This paradigm strengthens the client's data independence and raises researchers' extensive interest. Yu et al. [21] proposed an FL-based cooperative hierarchical caching scheme to address content popularity prediction's privacy issues in IoV-enabled fog networks. Chen et al. [22] focused on client selection to improve training efficiency in asynchronous FL with unstable connection in IoV. Zhao et al. [23] model the training and transmission latency of a novel FL paradigm combined with blockchain and adopt a duel deep Q-learning algorithm to minimize the system latency. To enhance the perception of IoV applications, Lim et al. [24] activated unmanned aerial vehicles and enabled an FL-based approach for their privacy-preserving collaboration. However, the existing researchers hardly pay attention to transportation issues with FL-empowered IoV.

When privacy problems are involved in vehicle motion control optimization, all data should be kept in a vehicle. To obtain better motion control, this study relaxes the privacy restriction to a local area, that is, a single-intersection area. Different intersections construct independent scopes. To achieve cooperative optimization among intersections, a Traffic-Aware Federated Imitation Learning framework for Motion Control (TAFI-MC) is proposed for acquiring vehicle motion control policies across different intersections. The main contributions of this study are summarized as follows.

- TAFI-MC framework is proposed to optimize motion control across multiple isolated unsignalized intersections cooperatively. This framework contains three parts: vehicle interactors, edge trainers, and one cloud aggregator;
- TAFI-MC integrates an IL algorithm to obtain a safety-oriented motion control policy, which trains the model with the experience from a set of collision avoidance rules;
- A loss-aware experience selection strategy is designed, which can reduce the communication overhead by extra computation. Depending on the reference loss, each interactor generates new experiences and decides whether to upload them.

The rest of this paper is organized as follows: The system architecture is presented in Section 2. The proposed TAFI-MC for vehicle motion control is detailed in Section 3. In Section 4, a loss-aware experience selection strategy is presented. The experimental results and discussions are provided in Section 5. Section 6 concludes this paper.

2. System Architecture

This section focus on a hierarchical network in an urban scenario, which includes one Cloud Aggregator (CA), tens of Edge Trainers (ETs), and hundreds of Vehicle Interactors (VIs), as shown in Figure 1. The CA is deployed in the remote cloud and connects to a set of ETs via a reliable backhaul link. These ETs are denoted by $\mathbb{ET} =$ $\{ET_1, ET_2, \ldots, ET_n, \ldots, ET_N\}$, where *N* is the number of ETs. Each ET *i* serves its wirelessly connected VIs, denoted by $\mathbb{VI} = \{VI_1, VI_2, \ldots, VI_i, \ldots, VI_{m(t)}\}$. Note that, under ET, the number of VI, m(t), varies over time *t*. ETs and VIs are equipped with adequate computing power. The local model at each ET is trained by combining uploaded information from various VIs. CA is used to create a global model by combining local information of different ETs. There are three assumptions to support our work, which is similar to related work [25], on the aspect of unsignalized intersection control. For simplification, this study focuses on longitudinal control. All vehicles proceed straight through the intersection area. All vehicles can measure kinetic information, adhere to the set acceleration, and communicate with adjacent nodes, i.e., ETs and adjacent VIs.

The longitudinal motion of vehicles is given by

$$x^{long}(t+1) = x^{long}(t) - v(t)T - \frac{1}{2}a(t)T^{2},$$

$$v(t+1) = v(t) + a(t)T$$
(1)

where x^{long} is the displacement, v and a are the velocity and acceleration, respectively, and T is the discrete-time step. The change of vehicle motion state depends on the input, i.e., acceleration, at the previous time step. Note that $x^{long}(t)$ represents the distance to the conflict point in intersection, so subtraction is used in the equation instead of addition.

This paper uses distributed decision-making for motion control. That is, each vehicle constructs its cyberspace and maps adjacent vehicles to cyber objects. As a result, vehicle i decides its action a_i based on its surroundings.

$$a_i = P(\overrightarrow{s_i} | \theta) \tag{2}$$

where $P(\cdot|\theta)$ is a θ -parameterized policy for decision-making. $\vec{s_i}$ is the state vector of vehicle *i* and $\vec{s_i} = \{s_i, \vec{s_{-i}}\}$. s_i is a state of the ego-vehicle *i*, including position, velocity, and acceleration. Moreover, $\vec{s_{-i}}$ is a vehicle set other than vehicle *i*. $|\vec{s_{-i}}|$, the number of vehicle set $\vec{s_{-i}}$ is defined by a selection scheme. To simulate traffic flow, the number of

arrived vehicles for entering each intersection during the period *t* is defined as $V_q(t)$. It follows a Poisson process with the parameter λ :

$$P(V_q(t) = g) = \frac{(\lambda t)^g}{g!} e^{-\lambda t}$$
(3)

where *g* equals the number of vehicles generated in a period *t*. The introduction of the Poisson process means that vehicles are dynamically created, which is similar to real traffic.

Figure 1 presents a three-layered federated deep learning architecture. The bottom layer contains VIs, a collection surrounding the information via vehicular communication. The middle layer includes several ETs equipped with computing servers and experience pools. The top layer has an aggregator, which forms global models with ETs' local models. The multiple connected VIs interact with the environment and individually upload experience data to the local experience pool on the corresponding ETs. Each ET first generates a local model and acquires the experience from connected VIs. Then, each ET uses the local computing ability to compute a local model based on the received data. Next, the ET sends the local model to the VIs for motion control and the CA for global model aggregation. Finally, the CA aggregates the models and distributes the global model back to each ET. The above steps are repeated until a convergent global model is achieved. The trained model in this work is specially developed to output the vehicles' accelerations in response to the contextual information surrounding the VI.



Figure 1. Federated Deep Learning Architecture for Motion Control.

3. Federated Imitation Learning Framework for Motion Control

The proposed TAFI-MC framework is elaborated in this section. First, Traffic-Aware Federated Learning is described. Then, we introduce a set of collision avoidance rules as a basis for further optimization. Finally, an IL algorithm for motion control is investigated using rules.

3.1. Traffic-Aware Federated Learning

FL enables collaborative training of a deep neural network model among ETs under CA's orchestration by storing the training data on each ET at intersections. With FL, privacy-related data can be kept within the scope of ET. After ET training, data is turned into a neural network model, which makes it hard to extract raw information. These privacy-related data include the vehicles' identity and position, inferring their destination, and drivers' identity. FL not only significantly reduces the vehicle's privacy risk but

also significantly reduces the communication overhead caused by centralized machine learning [26]. FL is enabled by multiple communication rounds (computing iteration). To conduct model training, *N* intersections with known traffic flows and the corresponding ETs are selected. The *N* ETs are indexed by *n*. Then, each ETs retrieves a global model from the CA and trains it with data collection from vehicles in the intersection area using its local model. Following the ET's local training, the updated weights and gradients are returned to the CA. Finally, the CA aggregates the models collected from vehicles to construct an updated global model. Furthermore, the final-trained models are distributed to the vehicles to ensure that they pass through the intersection.

The details of the designed FL iteration consists of the following steps:

- (1) *Model Distribution:* A set of ETs at intersections participate in FL training. The CA distributed the global model ω_r to ETs. The ET *n* trains the global model with local data for a new model ω_r^n . The index of communication rounds is represented by *r*.
- (2) Experience Upload: To improve motion control performance, each vehicle should consider other vehicles' states for inference. However, the efficiency of vehicular distributed training is significantly low because of the insufficient number of collected samples and non-uniform distribution. Under the above setting, it is essential to use centralized training and distributed execution [26]. Then, each vehicle interacts with the environment, i.e., other vehicles, and generates enormous experience data to upload. The corresponding ET uses this data to train the local model.
- (3) *FL Model Training*: The proposed FL's third step is to train the model by using local data uploaded by vehicles. Let $Exp = \{Exp_1, Exp_2, \dots, Exp_n, \dots, Exp_N\}$ represent the experience data stored in selected ETs. Exp_n denotes a local experience of the *n*th ET with a length d_n , $d_n = |Exp_n|$. *d* is the size of the entire data among the selected ETs. The goal of the FL is to minimize the loss function $L(\omega)$:

$$\min_{\omega} L(\omega) = \sum_{n=1}^{N} \frac{d_n}{d} L_n(\omega) \quad \text{where}$$

$$L_n(\omega) = \frac{1}{d_j} \sum_{j \in H_k} l_j(\omega), \qquad (4)$$

where $l_j(\omega)$ is a loss of the motion control on the *j*th experience batch in Exp_n with the parameters of model ω , and d_j is the experience number in the *j*th batch. *n* denotes the index of total selected ETs *N*. $L_n(\omega)$ represents the local loss function of ET *n*. Then, minimizing the weighted average of the local loss function $L_n(\omega)$ is equal to optimizing the loss function $l(\omega)$ of FL. With this step, the experience with sensitive information is transformed into a neural network model, which is hard to extract sensitive information from.

- (4) Upload Updated Model: The fourth step is to upload the local model ω_{t+1}^n from ETs to the CA. The communication overhead exceeds the computing overhead [27]. The model can be compressed before being uploaded to the CA to reduce communication overhead.
- (5) Weighted Aggregation: After ETs upload their models, the fifth step is to produce a new global model ω_{r+1} by computing a weighted sum of all received models ω_r^n . For the next training iteration, the newly generated global model is used. Federated Averaging (FedAVG), which is commonly used in FL, increases the proportion of local computing and decreases mini-batch sizes. In FedAVG, each ET adds computation by iterating the local updates $w_r^n \leftarrow w_r^n \eta \nabla L_n(w_r^n)$ multiple times before the aggregation step in the CA. To aggregate the model, the weighted averaging algorithm is implemented. The weights for parameter aggregation are determined by the traffic flow of each intersection, which is $\gamma_n = F_n/F$. F_n and F denote the traffic flow on

*n*th intersections and the flow sum of all intersections, respectively. The aggregate method can then be re-written as

$$w_{r+1} \leftarrow w_r - \eta \sum_{n=1}^N \gamma_n \frac{d_n}{d} w_{r+1}^n.$$
(5)

Selected intersections with a higher traffic flow contribute more and are given greater weight in model aggregation.

3.2. Imitation Learning for Motion Control

IL model for motion control is trained in the above FL framework. It is used to obtain the motion control ability from collision avoidance rules. IL and RL both depend on environment interaction. Unlike RL, which obtains the desired behaviors based on the hidden objectives, IL directly clones the desired behaviors. IL can overcome RL's highly uncertain initial state and the sparse reward, which can lead to an exploration trap. Therefore, this section explains how to imitate the end-to-end motion control policy using existing rules.

As shown in Figure 2, two modules are presented to support motion control policy acquisition in ET. The upper module is a set of collision avoidance rules to output expert experience, a deterministic action with fixed experience input. As the final motion control policy carrier, a deep neural network is continuously updated in the lower module. The network is updated with loss, which comes from the difference between the output action from the two modules. The proposed IL is shown in Algorithm 1.

Algorithm 1: Imitation learning for motion control.

- 1: Collision avoidance rules guide vehicles to make actions *a*
- 2: The state *s* and action *a* are recorded as expert experience (*s*, *a*) to upload to the corresponding ET's experience buffer
- 3: ET samples a batch of experience \overline{Exp} from the pool
- 4: Exp is simultaneously forwarded to two modules, collision avoidance rules, and deep neural network
- 5: The loss is calculated with the output of the two modules, $\pi_{\theta}(\vec{s_i})$ and a_i
- 6: The deep neural network is updated by minimizing the loss

$$L(\overrightarrow{Exp}) = \frac{1}{B} \sum_{i=1}^{B} |\pi_{\theta}(\overrightarrow{s_i}) - a_i|^2.$$
(6)

In Equation (6), \overrightarrow{Exp} is a batch of experience, and the batch size is *B*, given in the experiment part. $\pi_{\theta}(\cdot)$ is a θ -parameterized policy, and each iteration updates parameter θ . The state vector $\overrightarrow{s_i}$ including the state of current vehicle and surrounding vehicles, i.e., $\overrightarrow{s_i} = \{s_i, \overrightarrow{s_{i,1}}\} = \{s_i, \overrightarrow{s_{i,1}}, \cdots, \overrightarrow{s_{i,n}}\}$. *n* is the number of the closest vehicles to be considered into a state vector. Each state s_* includes position, velocity, and acceleration, i.e., $s_* = \{x^{long}, v, a\}$. The definition of closest vehicles relies on the cyber-lane, which is described in Section 3.3. The loss function can drive the deep neural network to produce safety-oriented strategies.



Figure 2. Imitation Learning for Motion Control.

3.3. Collision Avoidance Rules

In this section, the concept of Cyber-Lane (CL) is introduced to reconstruct vehicles' relationships. Vehicles in different trajectories travel in different Physical-Lanes (PLs), and these trajectories intersect at conflict points. As presented in Figure 3, lane-A is taken as an example. The generation of CL-A is based on PL-A, and vehicles on the conflict lane are projected to CL-A centered on the conflict point. In other words, the PL number is equal to the CL number. After being projected, $D1 \rightarrow A$ and $B1 \rightarrow A$ appear on CL-A. From the standpoint of CL, the position relationship of all vehicles is reorganized. Vehicle *A*2 appears between vehicles *B*1 and *D*1 on CL-A, and vehicle *A*2 treats vehicle *B*1 or *D*1, instead of vehicle *A*1.



Figure 3. Vehicle projection from a physical lane to a cyber lane.

The set of rules considers three factors, including space, time, and acceleration. Space is the first factor, which can directly determine whether the collision has occurred. As a second-order factor, time considers whether the vehicle will collide in the future. The acceleration indicates whether the collision will be avoided. In this paper, the above factors are quantified as Safety Value (*SV*).

The first factor is space. The safety value for space $SV_{i,s}$ is calculated as below,

$$SV_{j,s} = log((\frac{d_{j,nearest}}{\alpha_s})^{\beta_s}),$$
(7)

where $d_{j,nearest}$ denotes the distance between vehicle *j* and its nearest vehicle on the virtual lane. α_s normalizes $d_{j,nearest}$, and it can be treated as the expected headway distance. β_s increases the offset to improve the $log(\cdot)$ effect. The *SV* and nearest inter-vehicle space distance have a positive correlation.

The second factor is time. The SV for time $SV_{j,t}$ is calculated as below,

$$SV_{j,t} = \begin{cases} -\left[\frac{\alpha_t}{\tanh(-t_{j,nearest})}\right]^{\beta_t} & 0 < t_{j,nearest} < 1\\ 2 & otherwise \end{cases},$$
(8)

where $t_{i,nearest}$ denotes Time To Collision (TTC) between vehicle *j* and its nearest vehicle. The function $tanh(\cdot)$ is used to mark the nearby collision risk in the sensitive range, where $t_{i,nearest}$ is less than 1. TTC and *SV* have a negative correlation.

The third factor is acceleration. The SV for acceleration $SV_{j,acc}$ is calculated as below,

$$SV_{j,acc} = \lambda_{acc} \times acc_{j,front} \times \log\left(\min\left(\frac{d_{j,front}}{d_{threshold}},\alpha_{acc}\right)^{\beta_{acc}}\right),\tag{9}$$

where $d_{j,front}$ is the distance from vehicle *j* to its front vehicle, $acc_{j,f}$ is the acceleration of the vehicle in front of vehicle *j*, and $d_{threshold}$ is the space distance safety threshold. $min(\cdot)$ is used to control the range of $\frac{d_{j,front}}{d_{threshold}}$ within $[0, \alpha_{acc}]$. Discount factor λ_{acc} is introduced to limit the influence of acceleration in the calculation of *SV*.

The combination of *SV* is calculated as follows,

$$SV_{j} = Comb(SV_{j,s}, SV_{j,t}, SV_{j,acc}|SV_{max}, SV_{min})$$

= $clip((SV_{j,d} + SV_{j,t} + SV_{j,acc}), SV_{max}, SV_{min}),$ (10)

where $SV_{j,d}$, $SV_{j,t}$, and $SV_{j,acc}$ are defined above. To obtain a proper acceleration value in Equation (11), $clip(\cdot)$ is used to limit the maximum and minimum values. A larger SV_j indicates that the vehicle j is driving in a safer environment. Based on the above SV, the ego vehicle's action can be calculated as follows,

$$a_{exe} = \begin{cases} \left| \frac{SV}{\eta} \right| & d_f \le d_b \\ \frac{SV}{\eta} & d_f > d_b \end{cases}$$
(11)

where d_f is the distance to the vehicle in front, d_b is the distance to the vehicle behind, and η is used to convert safety value to action, i.e. ego vehicle's acceleration. The experimental results shown in Section 5 demonstrate that rules can achieve collision avoidance under different traffic flows.

4. Loss-Aware Experience Selection Strategy

In the setting of the proposed IL in Section 3, the experience, i.e., (s, a) tuple, is generated by each vehicle and uploaded to ETs for training at 10 Hz. This will consume many communication resources when the number of vehicles near an intersection is enormous. This section introduces computing for communication, where additional computation is performed to reduce communication overhead. In this section, the extra computation is placed on vehicles and edge nodes. Vehicles calculate the loss and compare it to thresholds given by edge nodes. Edge nodes produce a threshold for loss comparison. Therefore, combined with the concept of computing for communication, a loss-aware experience selection strategy is proposed to discard experience that helps in model training.

As displayed in Figure 4, the proposed strategy is applied between vehicle interactors and edge trainers. When a vehicle enters an intersection area, it requests edge trainers for the most recent model and threshold. Then, using vehicular collision avoidance rules, all vehicles interact with the environment and generate experience (s, a). The vehicle acts with other vehicles and outputs action loss between the rules and the model. The loss is compared with loss threshold *Th*. If the action loss is larger than *Th*., the corresponding experience (s, a) can be uploaded to the edge trainers, and vice versa. Finally, the acquired experiences enable the edge trainers to perform IL training and output a new threshold. The threshold is calculated on ETs as follows,

$$Th. = sort(\overrightarrow{Exp}, "loss")[p \times B]["loss"]$$
(12)

where Exp is a batch of experience for training, and its size is *B*. The function *sort*(·, "*loss*") sorts experiences in ascending order of loss value. *p* is the discard rate. According to Equation (12), the threshold is the $(p \times B)$ th smallest loss in the experience batch. Because the experience will be partly discarded, the communication overhead is reduced.



Figure 4. Loss-aware experience selection strategy.

CMI

Compare Loss

5. Results and Discussion

5.1. Simulation Settings

Calculate Loss

The proposed algorithm is trained and evaluated in a self-designed intersection motion control platform, developed with Python 3.5. The platform contains four of the same intersections, but they differ in traffic flow rates. Considering that different intersections have different traffic flow characteristics, isolated intersections with different traffic flows are set up to verify the motion control brought by TAFI-MC. In this experiment, the traffic flows are 300/900/1500/2100 veh/lane/hour, where veh means the number of vehicles. According to different traffic flow rates, vehicles are generated using the Poisson process and allowed to go straight without steering. The related motion control parameters are listed in Table 1. All source codes about the proposed TAFI-MC framework are provided on GitHub (https://github.com/shogun2015/TAFI-MC (accessed on 3 December 2021)).

Table 1. Experimental Parameters.

Parameter	Value
Simulator	
Lane length (m)	150
Vehicle size (m)	2
Velocity (m/s)	[6,13]
Initial velocity (m/s)	10
Acceleration (m/s^2)	[-3,3]
Discrete-time step T (s)	0.1
Safety Value	
Space normalization factor α_s	10
Space exponetial factor β_s	10
Time linear factor α_t	1.5
Time exponetial factor β_t	2
Acceleration exponetial factor α_{acc}	1.5
Acceleration exponetial factor β_{acc}	12
Acceleration linear factor λ_{acc}	0.2
Safety value upper bound SV_{max}	20
Safety value lower bound SV_{min}	-20
Conversion factor η	3
Fusion factor ω	0.2
Weighting factor ε	0.5
Vehicle Selection	
Number of the closest vehicle <i>n</i>	5

The starting point of this study is to balance model performance and privacy protection. Therefore, this study allows ETs to collect vehicle information in their corresponding intersections to complete IL training. In other words, each IL training only handles a single flow-rate traffic. In the proposed schemes, the neural network (NN) is used to imitate the collision-free rules by minimizing the action loss between the NN and the rules. There is only one NN, containing three dense layers and two normalization layers, and ReLU is used as the activation function in the hidden layers. The output layer is activated by $tanh(\cdot)$. To fit the acceleration range, the NN output is multiplied by 3. The complete hyperparameters are listed in Table 2. However, because of poor interpretability and limited safety performance of end-to-end NN inference, a weighted operation is added below,

$$a_{exe} = \varepsilon \times a_{NN} + (1 - \varepsilon) \times a_{rule}, \tag{13}$$

where ε is a factor for smoothing the NN output a_{NN} with a rule output a_{rule} to ensure safe driving. In the following experiment results, *Model* denotes action output using NN only, *Model*+*Rule* represents mixed output using NN and rules.

Table 2. Parameters for Neural Networks.

Parameter	Value			
Discounted factor γ	0.8			
Batch Size B	48			
Soft update factor $ au$	0.99			
Episode	50			
Learning rate	0.001 ightarrow 0			
Optimizer	Adam			
Network Architecture				
Dense layer 1#	64			
Dense layer 2#	64			
Dense layer 3#	1			

5.2. Metric

Three indicators are chosen to comprehensively evaluate the performance of the proposed motion control methods at intersections, i.e., collision rate, average jerk, and average velocity.

Collision rate is the first metric to evaluate the motion control safety, which is shown below,

$$r_{collision} = \frac{n_{collision}}{N_{veh}},\tag{14}$$

where $n_{collision}$ is the number of collisions, and N_{veh} is the total number of vehicles. A larger $r_{collision}$ means the motion control algorithm is incapable of achieving collision avoidance. The algorithm in this study is designed to reduce the metric value to 0.

Average jerk is an important metric to evaluate motion control comfort. As presented in [28,29], the average jerk can be defined as below,

$$J_{avg} = \frac{1}{N_{veh}} \sum_{i=1}^{N_{veh}} \sum_{t}^{N_{i}} j_{i,t}^{2},$$
(15)

where $j_{i,t}$ is the *i*th vehicle's jerk. The jerk is defined by $j_{i,t} = a_{i,t}$, and $a_{i,t}$ is the *i*th vehicle's acceleration at the time step *t*. A larger J_{avg} indicates more frequent or sharp acceleration and deceleration, resulting in more severe driving discomfort.

While ensuring a low collision rate, average velocity is a metric that must be improved. It can be defined as below,

$$v_{avg} = \frac{1}{N_{veh}} \sum_{i=1}^{N_{veh}} \frac{l_{lane}}{t_i},$$
(16)

where t_i is the *i*th vehicle's travel time and l_{lane} is the lane length, which is given in Table 1. A larger v_{avg} represents vehicles with the proposed algorithm that will drive faster, resulting in high throughput for the transportation system.

5.3. Discussion

The entire experiment is divided into two parts. In the first part, the benchmark Reinforcement Learning (RL), the proposed IL, and TAFI-MC are evaluated from the three metrics: collision rate, average jerk, average velocity, as described in Section 3.3. The IL and RL are trained in four traffic flows (i.e., 300/900/1500/2100 veh/lane/hour), and verified in seven traffic flows (i.e., 300/600/900/1200/1500/2100 veh/lane/hour). The set of collision avoidance rules described in Section 3.3 is represented by a black line in Figures 5a, 6a, and 7a. TAFI-MC is evaluated with two aggregation methods (same-proportion and traffic-aware). Same-proportion means the four models have the same weight, whereas traffic-aware means the weight is 1 : 3 : 5 : 7. The second part verifies the performance of the proposed loss-aware experience selection strategy with different discard factors *p*.





(c) The comparison of collision rate with TAFI-MC

Figure 5. The comparison of collision rate *r*_{collision}.

Figure 5 depicts the comparison of collision rate with the three algorithms, IL, RL, and TAFI-MC. In Figure 5a, the black line remains at zero collision, which means that the rules have good collision avoidance ability at all experimental flow rates. This is because the rules are safety oriented and avoid all collisions from the perspective of position, speed, and acceleration. In addition, the results demonstrate that the proposed IL scheme can

help the model learn collision avoidance from the rules. Only in high flow evaluation environment will the model trained in low flow produce some collisions. This is because, in the low traffic flow, the experience samples come from large inter-vehicle distances. The model trained by this experience has difficulties guiding a vehicle to handle a small inter-vehicle distance. As shown in Figure 5b, the single traffic flow training makes RL show unacceptable collision rates, because RL is more dependent on interaction samples, which is low efficiency. RL's low safety performance exceeds the correction ability of rules. In Figure 5c, traffic-aware model aggregation method outperforms the same-proportion method. The traffic-aware method does not need rules correction. To sum up, IL can obtain stronger collision avoidance ability than RL to greatly reduce the collision rate, and TAFI-MC with the proposed traffic-aware model aggregation can enhance collision avoidance.



(a) The comparison of average jerk with IL

(b) The comparison of average jerk with RL



(c) The comparison of average jerk with TAFI-MC

Figure 6. The comparison of average jerk of J_{avg} .

Figure 6 shows the comparison of average jerks with the three algorithms. In Figure 6a, collision avoidance rules show the highest average jerk, because rule-making inevitably adopts the threshold trigger mechanism. Only considered states can be mapped to actions. Compared with rules, IL significantly reduces jerks than rules. The model training process helps IL complete the state-action mapping by gradually approaching the rules with a low learning rate. Note that, although rules can reduce collision rates, their threshold trigger mechanism will increase jerks inevitably. As discussed in the collision rate, the model trained in high traffic is also slightly insufficient when it comes to dealing with low traffic. This is mainly reflected in that the high-traffic model has higher average jerk than the low-traffic model. The performance of RL, in Figure 6b, is similar to IL. This demonstrates that the control policies based on interaction experiences have similar effects on driving comfort. As shown in Figure 6c, TAFI-MC with traffic-aware model aggregation can further reduce average jerks by up to 41.37% than IL trained models at any single traffic flow.

Taking a comprehensive view at the three figures in Figure 7, we can see that the three algorithms achieve high average velocity, which also means high traffic efficiency and throughput. With the increase in traffic flow, the average velocity has decreased slightly. This is because more vehicles rise the collision probability of vehicles, and the collision avoidance motions of vehicles reduce the average velocity. In Figure 7a,b, models trained under sparse traffic flows show relatively high velocity, because a small number of vehicles makes it hard to cause the collision avoidance motion of vehicles. In Figure 7c, FL aggregates the gap of different models' motion control. Furthermore, traffic-awareness brings slightly lower velocity, but ensures zero collision rate. In combination with Figures 5–7, the reduction of collision rates inevitably leads to the increase of average jerks. The most obvious ones are the models trained in the most sparse traffic flow, i.e., 300 veh/lane/hour. Fortunately, FL introduced by TAFI-MC makes up for this deficiency and keeps different metrics at better values.



(a) The comparison of average velocity with IL (b) The comparison of average velocity with RL



Figure 7. The comparison of average velocity v_{avg} .

In Figure 8, the convergent loss curves are represented by solid lines and the nonconvergent curves are represented by dotted lines. The results demonstrate that, with the number of discarded experiences increasing, the IL model is more difficult to converge. It shows the loss curves with different discard factors p. When p is no more than 10% and the training step exceeds 4000, the trained models can be convergent and stable. It can be easily found that the models converge and stabilize from 6000 training steps, thus only the communication overheads before convergence is counted. In addition, the loss curves peak at about 2000 training steps. Since that, losses and models have been gradually stable. In the following analysis, only the convergent model (i.e., models with p = 1%, 2%, 5%, 10%) is considered.



Figure 8. The loss value with different discard factor *p*.

In Figure 9, the cumulative communication reduction rate is presented. A higher p brings less communication overhead. At the beginning of model training, the VIs' model parameters are random. As a threshold, the reference loss discards more experience, which leads to a greater reduction in communication overhead. After the loss curves peak at 2000 training steps, the communication reduction rates gradually decrease with the training progress. When the model converges, the cumulative communication overhead can be reduced by 0.44%, 1.65%, 5.6%, and 12.80% with different discard factors.



Figure 9. Cumulative communication reduction rate with a different discard factor *p*.

According to the performance in Table 3, the reduction of experience did not lead to performance degradation. In short, the proposed loss-aware experience selection strategy reduces the communication overhead while ensuring model performance.

Discard Factor		1%	2%	5%	10%
Collision Rate	Model	36%	39%	44%	42%
r _{collision}	Model+Rule	0%	0%	0%	0%
Average Jerk	Model	21.55	13.08	11.11	23.85
Javg	Model+Rule	137.15	113.23	108.83	130.83
Average Velocity	Model	12.22	12.06	12.06	12.24
v _{avg}	Model+Rule	12.21	12.21	12.19	12.26

Table 3. Performance with discard factors.

6. Conclusions

A Traffic-Aware Federated Imitation Learning Framework for Motion Control was proposed to optimize motion control across multiple isolated unsignalized intersections. The framework contains vehicle interactors, edge trainers, and a cloud aggregator. Data privacy restriction is relaxed from vehicle interactors to edge trainers at intersections to balance privacy-preserving and motion control. The framework integrated a traffic-aware model aggregation for intersections with different traffic flows. Then, an IL algorithm was proposed for action cloning from a set of collision avoidance rules to improve the safety capability of end-to-end learning. Furthermore, this paper explored a loss-aware experience selection strategy to reduce the communication overhead through additional computation on interactors and trainers. The extensive experiment revealed that the proposed IL algorithm could achieve collision avoidance and improve the driving comfort. The TAFI-MC framework meets the privacy demand and further improves driving comfort. The proposed experience selection strategy can reduce the communication overhead while ensuring convergence.

Our future work will focus on the modeling and theoretical analysis of the relationship between the interactors and trainers in terms of communication and model training. Based on this analysis, we believe it will help accelerate model training while significantly reducing the communication overhead.

Author Contributions: Conceptualization, T.W.; methodology, T.W. and M.J.; software, Z.Y. and Y.H.; validation, M.J., Z.Y. and Y.H.; investigation, T.W. and X.L.; resources, L.Z.; data curation, Z.Y.; writing—original draft preparation, T.W.; writing—review and editing, L.Z.; visualization, Z.Y. and Y.H.; supervision, L.Z.; project administration, T.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China (2016YFB0100902).

Data Availability Statement: The code is open-sourced at https://github.com/shogun2015/TAFI-MC (accessed on 3 December 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- FL Federated Learning
- IoV Internet of Vehicles
- IL Imitation Learning

References

- 1. Chen, Y.; Lu, C.; Chu, W. A Cooperative Driving Strategy Based on Velocity Prediction for Connected Vehicles With Robust Path-Following Control. *IEEE Internet Things J.* 2020, *7*, 3822–3832. [CrossRef]
- Xu, Y.; Li, D.; Xi, Y. A Game-Based Adaptive Traffic Signal Control Policy Using the Vehicle to Infrastructure (V2I). *IEEE Trans. Veh. Technol.* 2019, 68, 9425–9437. [CrossRef]

- 3. Khayatian, M.; Mehrabian, M.; Andert, E.; Dedinsky, R.; Choudhary, S.; Lou, Y.; Shirvastava, A. A Survey on Intersection Management of Connected Autonomous Vehicles. *ACM Trans.-Cyber-Phys. Syst.* **2020**, *4*, 48:1–48:27. [CrossRef]
- 4. Stryszowski, M.; Longo, S.; Velenis, E.; Forostovsky, G. A framework for self-enforced interaction between connected vehicles: Intersection negotiation. *IEEE Trans. Intell. Transp. Syst.* 2020, 22, 6716–6725. [CrossRef]
- 5. Perronnet, F.; Buisson, J.; Lombard, A.; Abbas-Turki, A.; Ahmane, M.; Moudni, A.E. Deadlock Prevention of Self-Driving Vehicles in a Network of Intersections. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 4219–4233. [CrossRef]
- 6. Guan, Y.; Ren, Y.; Li, S.E.; Sun, Q.; Luo, L.; Li, K. Centralized cooperation for connected and automated vehicles at intersections by proximal policy optimization. *IEEE Trans. Veh. Technol.* **2020**, *69*, 12597–12608. [CrossRef]
- Wu, T.; Jiang, M.; Zhang, L. Cooperative Multiagent Deep Deterministic Policy Gradient (CoMADDPG) for Intelligent Connected Transportation with Unsignalized Intersection. *Math. Probl. Eng.* 2020, 2020, 1820527. [CrossRef]
- 8. Jiang, M.; Wu, T.; Wang, Z.; Gong, Y.; Zhang, L.; Liu, R.P. A Multi-intersection Vehicular Cooperative Control based on End-Edge-Cloud Computing. *arXiv* 2020, arXiv:2012.00500.
- Guo, W.; Tian, W.; Ye, Y.; Xu, L.; Wu, K. Cloud Resource Scheduling with Deep Reinforcement Learning and Imitation Learning. IEEE Internet Things J. 2020, 8, 3576–3586. [CrossRef]
- Huo, Y.; Tao, Q.; Hu, J. Cooperative Control for Multi-Intersection Traffic Signal Based on Deep Reinforcement Learning and Imitation Learning. *IEEE Access* 2020, *8*, 199573–199585. [CrossRef]
- Yang, B.; Zhang, J.; Shi, H. Interactive-Imitation-Based Distributed Coordination Scheme for Smart Manufacturing. *IEEE Trans. Ind. Inform.* 2020, 17, 3599–3608. [CrossRef]
- 12. Riedmaier, S.; Ponn, T.; Ludwig, D.; Schick, B.; Diermeyer, F. Survey on scenario-based safety assessment of automated vehicles. *IEEE Access* 2020, *8*, 87456–87477. [CrossRef]
- 13. Tesla. Tesla AI Day. 2021. Available online: https://www.youtube.com/watch?v=j0z4FweCy4M&ab_channel=Tesla (accessed on 3 December 2021).
- 14. Luo, P.; Yu, F.R.; Chen, J.; Li, J.; Leung, V.C. A Novel Adaptive Gradient Compression Scheme: Reducing the Communication Overhead for Distributed Deep Learning in the Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 11476–11486. [CrossRef]
- Shi, S.; Wang, Q.; Chu, X.; Li, B.; Qin, Y.; Liu, R.; Zhao, X. Communication-efficient distributed deep learning with merged gradient sparsification on GPUs. In Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications, Toronto, ON, Canada, 6–9 July 2020; pp. 406–415.
- 16. Sattler, F.; Wiedemann, S.; Müller, K.R.; Samek, W. Robust and communication-efficient federated learning from non-iid data. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 3400–3413. [CrossRef]
- 17. Schaul, T.; Quan, J.; Antonoglou, I.; Silver, D. Prioritized Experience Replay. In Proceedings of the 4th International Conference on Learning Representations, ICLR 2016, San Juan, PR, USA, 2–4 May 2016; pp. 1–21.
- Messaoud, S.; Bradai, A.; Ahmed, O.B.; Quang, P.T.A.; Atri, M.; Hossain, M.S. Deep federated q-learning-based network slicing for industrial iot. *IEEE Trans. Ind. Inform.* 2020, 17, 5572–5582. [CrossRef]
- 19. Mun, H.; Lee, Y. Internet Traffic Classification with Federated Learning. *Electronics* 2021, 10, 27. [CrossRef]
- 20. Li, Z.; Liu, J.; Hao, J.; Wang, H.; Xian, M. CrowdSFL: A Secure Crowd Computing Framework Based on Blockchain and Federated Learning. *Electronics* **2020**, *9*, 773. [CrossRef]
- Yu, Z.; Hu, J.; Min, G.; Wang, Z.; Miao, W.; Li, S. Privacy-Preserving Federated Deep Learning for Cooperative Hierarchical Caching in Fog Computing. *IEEE Internet Things J.* 2021, 1–10. [CrossRef]
- Chen, Z.; Liao, W.; Hua, K.; Lu, C.; Yu, W. Towards asynchronous federated learning for heterogeneous edge-powered internet of things. *Digit. Commun. Netw.* 2021, 7, 317–326. [CrossRef]
- 23. Zhao, N.; Wu, H.; Yu, F.R.; Wang, L.; Zhang, W.; Leung, V.C. Deep Reinforcement Learning-Based Latency Minimization in Edge Intelligence over Vehicular Networks. *IEEE Internet Things J.* **2021**. [CrossRef]
- Lim, W.Y.B.; Huang, J.; Xiong, Z.; Kang, J.; Niyato, D.; Hua, X.S.; Leung, C.; Miao, C. Towards federated learning in uav-enabled internet of vehicles: A multi-dimensional contract-matching approach. *IEEE Trans. Intell. Transp. Syst.* 2021, 22, 5140–5154. [CrossRef]
- 25. Bian, Y.; Li, S.E.; Ren, W.; Wang, J.; Li, K.; Liu, H.X. Cooperation of multiple connected vehicles at unsignalized intersections: Distributed observation, optimization, and control. *IEEE Trans. Ind. Electron.* **2019**, *67*, 10744–10754. [CrossRef]
- 26. Konecný, J.; McMahan, H.B.; Yu, F.X.; Richtárik, P.; Suresh, A.T.; Bacon, D. Federated Learning: Strategies for Improving Communication Efficiency. *arXiv* 2016, arXiv:1610.05492.
- McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, Fort Lauderdale, FL, USA, 20–22 April 2017; Volume 54, pp. 1273–1282.
- Zhang, Y.; Malikopoulos, A.A.; Cassandras, C.G. Decentralized optimal control for connected automated vehicles at intersections including left and right turns. In Proceedings of the 56th IEEE Annual Conference on Decision and Control, CDC 2017, Melbourne, Australia, 12–15 December 2017; pp. 4428–4433.
- Katriniok, A.; Kojchev, S.; Lefeber, E.; Nijmeijer, H. Distributed scenario model predictive control for driver aided intersection crossing. In Proceedings of the 2018 European Control Conference (ECC), Melbourne, Australia, 12–15 December 2017; pp. 1746–1752.