Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9665421, 13 pages
https://doi.org/10.1155/2022/9665421

Research Article

A Deep Coordination Graph Convolution Reinforcement Learning for Multi-Intelligent Vehicle Driving Policy

Huaiwei Si, Guozhen Tan, and Hao Zuo

School of Computer Science and Technology, Dalian University of Technology, No. 2 Linggong Road, Ganjingzi District, Dalian City, Liaoning Province 116024, China

Correspondence should be addressed to Guozhen Tan; 18310908958@163.com

Received 10 May 2022; Accepted 12 June 2022; Published 28 June 2022

Academic Editor: Chia-Huei Wu

With the growing up of Internet of Things technology, the application of Internet of Things has been popularized in the field of intelligent vehicles. Therefore, more artificial intelligence algorithms, especially DRL methods, are more widely used in autonomous driving. A large number of deep reinforcement learning (RL) technologies are continuously applied to the behavior planning module of single-vehicle autonomous driving in early. However, autonomous driving is an environment where multi-intelligent vehicles coexist, interact with each other, and dynamically change. In this environment, multiagent RL technology is one of the most promising technologies for solving the coordination behavior planning problem of multivehicles. However, the research related to this topic is rare. This paper introduces a dynamic coordination graph (CG) convolution technology for the cooperative learning of multi-intelligent vehicles. This method dynamically constructs a CG model among multiple vehicles, effectively reducing the impact of unrelated intelligent vehicles and simplifying the learning process. The relationship between intelligent vehicles is refined using the attention mechanism, and the graph convolution RL technology is used to simulate the message-passing aggregation algorithm to maximize the local utility and obtain the maximum joint utility to guide coordination learning. Driving samples are used as training data, and the model guided by reward shaping is combined with the model of the free graph convolution RL method, which enables our proposed method to achieve high gradualness and improve its learning efficiency. In addition, as the graph convolutional RL algorithm shares parameters between agents, it can easily build scales that are suitable for large-scale multiagent systems, such as traffic environments. Finally, the proposed algorithm is tested and verified for the multivehicle cooperative lane-changing problem in the simulation environment of autonomous driving. Experimental results show that our proposed method has better value function representation in that it can learn better coordination driving policies than traditional dynamic coordination algorithms.

1. Introduction

Autonomous driving, regarded as a cognitive system, is composed of the following three main models: perception, planning, and control [1, 2]. The various models of this cognitive system comprise many methods, each of them describing the subcomponents of those models and the interaction interfaces between the models [3]. The planning model can be divided into route planning, behavior planning, and motion planning [4]. An increasing number of research has used artificial intelligence in recent years to solve the behavior planning problem of autonomous driving [5], especially since the deep reinforcement learning (DRL) method has achieved great success [6], and many researchers have applied DRL to such a problem [7]. Most of the studies have applied the single-agent reinforcement learning (RL) method to the autonomous driving environment [8–10]. For example, the classic deep Q-network (DQN) method is applied to automatic driving to solve the lane-changing problem [11], and the actor–critic method is applied to the behavioral decision-making problem of automatic driving [12]. Most of the work has focused on the behavior decision making of single-intelligent vehicles. Although some of them have considered other road elements to predict their behavior [13], the goal had been to learn a decision-making method of single-intelligent vehicles, and the decision

making had not considered the integrated coordinated decision making of multi-intelligent vehicles.

However, the future scheme will likely involve intelligent transportation systems with multi-intelligent vehicles. Autonomous vehicles can obtain more information about other vehicles. If such vehicle types can coordinate with other intelligent vehicles; then, they can drive more safely and efficiently. For example, the interaction and coordination between intelligent vehicles and surrounding vehicles can help intelligent vehicles to understand road traffic information, the location of other vehicles, or the different behavior plans of other vehicles. Consequently, intelligent vehicles can make coordinated decisions that are conducive to the overall road situation and ensure a safer and more efficient intelligent transportation system. Multiagent DRLs typify a good method of solving the decision-making problem of multivehicle coordination driving [14]. However, the studies on multi-intelligent vehicle coordination methods at present are few.

In the multi-intelligent vehicle environment, the multiagent RL (MARL) [15] method can be used to learn the coordination policy of intelligent vehicles. However, large numbers of intelligent vehicles and dynamically changing environments may both complicate the interaction relationship in the policy learning process. Consequently, simplifying the relationship between agents in the learning process has gradually become a vital research field [16]. In a general multiagent environment, predefined rules are usually used to abstract the relationship between agents [17]. However, with the increasing number of agents and the growing complexity of environments, accurately defining the relationship between agents only by using predesigned rules has become increasingly difficult [16]. Some researchers have used the soft attention mechanism to calculate the importance distribution of each agent to its neighboring agents [18]. Although the attention mechanism can be used to learn the interaction between agents in the graph convolutional RL method, the output value of the softmax function is a relative value [19]. Consequently, cooperative agents unnecessarily obtain important weights, and truly modeling the relationship between agents becomes impossible. In addition, the softmax function usually generates extremely small but nonzero probability values that are assigned to irrelevant agents, thus, weakening the degree of attention that should have been given to cooperative agents. Especially in multi-intelligent vehicle environment, when the driving distance between vehicles is small and the vehicle density is large, as well as because the driving behavior of vehicles affects each other, the coordination between intelligent vehicles is particularly important for improving vehicle safety and traffic efficiency.

For traffic environments, such as highways, our previous work proposed a multivehicle coordination method based on a dynamic collaboration graph [20]. We use the safety field model between vehicles to dynamically construct the cooperative relationship between vehicles and use the multiagent learning method to learn the decision-making strategy of multivehicle cooperative driving. However, in the process of learning, we use the variable elimination method (VE) to solve the global utility, which needs to specify some rules artificially, which is contrary to the purpose of agent self-learning. Therefore, we use a graph neural network combined with reinforcement learning, which is a method of autonomous discovery and collaborative utility.

Based on the previous multivehicle coordination learning strategy decision [20], we have made the following contributions. (1) Further, the security field model is combined with the attention mechanism of the graph model and graph neural network, and the security field model is used as the hard attention mechanism of the graph model to dynamically construct the collaborative relationship. (2) The attention mechanism is used to learn the interaction weights between the explicit CGs. (3) The graph convolution process is used to simulate the belief propagation algorithm and solve the overall maximum utility, which is subsequently used to guide the intelligent vehicle to learn the coordination policy. (4) Existing expert knowledge is used to initially discover the coordination rules between intelligent vehicles and, on this basis, further learn coordination driving behavior policies. Moreover, in view of determining the effectiveness of the method, a set of scenarios involving 5, 8, and 11 vehicles are verified in a highway simulation environment. We conduct multivehicle training in an open-source simulation environment. Our method can get higher safety rewards and driving speed when multiple vehicles drive together and have scalability.

2. Related Work

In the studies about autonomous driving, many research institutions and scientific research teams have used artificial intelligence methods to enable intelligent vehicles to learn autonomously and promote the intelligent development of autonomous vehicles [1]. Among them, RL is an unsupervised learning method that can learn a policy based on real-time feedback, and it is widely used in the field of intelligent vehicle driving policy learning [4]. The RL method treats vehicle as an agent and learns driving policies through interaction with the environment [21]. The interaction process is a Markov decision process (MDP). Loiacono et al. [22] used traditional reinforcement learning to train autonomous vehicles to learn driving strategies in the simulation environment. Guo and Wu [23] used the approximate function combined with the policy gradient method to achieve good results in the racing game environment. In recent years, the DRL method that combines deep learning and RL has greatly promoted the application of RL in more complex driving environments. Some researchers combine driving rules with RL to train driving strategies [24]. Talpaert et al. [25] used DRL to learn in real-world simulation. In a simple autonomous driving scenario, Chae et al. [26] used the DRL method to make the autonomous vehicle learn how to brake. Belletti et al. [27] proposed a multiobjective vehicle merging strategy. Makantasis et al. [28] proposed a Q-mask DRL method to learn highway driving policies. In other studies, the DRL method was used to train autonomous vehicles to learn a safety policy in a variety of scenarios [29]. A hierarchical DRL framework was proposed to help

vehicles focus on surrounding vehicles and learn a smooth driving policy [30]. The proximal policy optimization was applied to control autonomous driving and subsequently to actual vehicles [31]. The other studies [32–34] introduced important research aspects pertaining to DRL in autonomous driving.

Behavior planning is one of the most concerned fields in automatic driving [35]. This aspect can make autonomous vehicles drive safely and efficiently. Many research on behavior planning has increasingly applied RL technology [36]. Alizadeh et al. [37] trained the DRL agent to control the transformation policy of intelligent vehicles in a simulated environment. Chen et al. [30] designed a hierarchical DRL algorithm to learn the lane-changing behavior in a dense traffic environment. Wang et al. [38] proposed a Q-learning method for automatic lane changes in highway environments. Yuan et al. [39] used various excitation mechanisms to learn different lane-changing policies in highway environments. Wang et al. [40] proposed a Q-learning method based on the dense microsimulation to learn lane changes in highways. Bey et al. [41] learned the tactical behavior planning of intelligent vehicles by predicting the characteristics of other vehicles. Sefati et al. [42] proposed an RL method to learn the tactical behavior planning of intelligent vehicles in urban scenes under uncertain conditions, in which the intentions of surrounding road users are taken into account in this method.

The above research shows that artificial intelligence methods, especially DRL, have been widely used in the field of automatic driving, but at present, there are few scenarios in which multivehicle cooperation is considered. These DRL methods mainly study the driving strategy of a single vehicle, while ignoring the interaction and coordination between multiple vehicles [4]. Obviously, the benefits of applying single agent learning method directly to multivehicle environment may be limited. Other methods propose graph theory as an abstract model of vehicle interaction and formation [43], but they mainly focus on formation and signal. Although some researchers have abstracted the cooperative relationship between multiple vehicles by using cooperative graph, they are only based on the relative position or initialization sequence number between vehicles [20, 44]. In the process of learning, they could only consider the local joint utility, but the individual utility is ignored, thus affecting the results of coordination learning.

3. Related Technologies

3.1. Markov Decision Processes and Reinforcement Learning. The nature of intelligent vehicle decision making is a random process according to the environment. Markov decision processes or MDPs are an important stochastic decision model of sequential decision making [45], which is the basis theoretical of reinforcement learning algorithm.

Define $\{X_n\}(n = 0, 1, 2, \cdots)$ is nonnegative integer random values, where $n \geq 0$, and nonnegative integer sequence: $i_0, i_1, \cdots, i_n$ and j, constant has established: $P\{X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \cdots, X_1 = i_1, X_0 = i_0\}$. Where $\{X_n\}(n = 0, 1, 2, \cdots)$ is discrete time Markov chains, for any i and j constant with: $P\{X_{n+1} = j | X_n = i\} = P\{X_1 = j | X_0 = i\}$. The Markov chains are homogeneous and independent increments. We consider X, it as a random state, and state transition function is independent of the state of history; this property is significant that it should be satisfied when solving the engineering problems; in this paper, the Markov chains are homogeneous chains.

The Markov chains, where $\{X_n\}(n = 0, 1, 2, \cdots)$ is state space in S, i and j are belong to S, the states i after n step to the states j the transition probability is ${p_{ij}}^n = p\{X_n = j | X_0 = i\}$, the probability p represents when n the value is 1. Describe the movement's influence on the state transition in MDPs. The MDPs are defined as a 5-tuble $\{S, A, r, P, \eta\}$, where S is discrete or continuous state space, A is discrete or continuous action space, $r : S \times A \longrightarrow R$ is reward function, η is to be optimistic objective function and satisfied the following Markov property, that is, $\forall i, j \in S, a \in A$ and $n \geq 0$. The $P(i, a, j)$ is transition probability where state i after performing an action a turn to state j, $r(i, a, j)$ is the reward function where state i after performing an action a turn to state j. The decision making objective is η, $\eta = E[\sum_{t=0} \gamma^t r_t]$ is total expected return function, where $E(\cdot)$ is mathematic expectation, $\gamma \in [0, 1)$ is the delayed parameter that represents a discount on the rate of return over time, and r_t is a immediately reward for performing an action in a state at t moment.

Reinforcement learning by optimizing the object of value function or policy to realize the control optimization in finally. Suppose S_t and A_t states and actions set at t moment. π_t is a policy at t moment, $\pi = (\pi_0, \pi_1, \cdots)$ is MDPs action policy set, the action policy set as a mapping $\pi : S \longrightarrow A$. The decision objective can be maximized with any initial state. The MDPs state value functions:

$$V^{\pi}(s) = E^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s\right], \tag{1}$$

where $E^{\pi}(\cdot)$ is mathematical expectation of strategic π. $V^{\pi}(s)$ is the total expected return of policy π and a discount on the subsequent state.

Define the value of a state S under policy π, $V()$ is the expected return when starting in S and following π. The concept of value function is introduced to optimize a policy. The MDP action value function is defined as

$$Q^{\pi}(s, a) = E^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s, a_0 = a\right], \tag{2}$$

where $E^{\pi}(.)$ is mathematical expectation π. In our approach, state transition probability and reward return model are unknown, but we can observe completely state information for each of intelligent vehicle. This paper we applied is known as Q-learning [46]. Q-learning makes an expected discount on future rewards. The state s takes an action a at t moment, the R is reward value when the state s at t moment, and they are all observed. The s' is next state. Synchronization of Q values is updated as $Q(s, a) \coloneqq Q(s, a) + \alpha$

$[R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)]$. Where $\alpha \in (0,1)$ is a learning rate, Q-learning converges to an optimal $Q^*(s,a)$ value, if all state action pairs have been detected with a reasonable exploring strategy.

3.2. Single-Intelligent Vehicle MDP and RL. In this paper, we simulated the motorway scenarios. Figure 1 shows the surroundings the intelligent vehicle perceived, which is showed how constructing the coordination graph. Take the no. 0 intelligent vehicle as an example, where d_1 is the distance between the nearest vehicle and in front of no. 0 intelligent vehicle in carriageway, v_1 is velocity, and a_1 is acceleration. d_2 is distance between the nearest vehicle and in the rear of no. 0 intelligent vehicle in carriageway, and v_2 and a_2 are, respectively, for velocity and acceleration. d_3 is distance between the nearest vehicle and in front of no. 0 intelligent vehicle in overtaking lane. v_3 and a_3 are, respectively, for velocity and acceleration. d_4 is distance between the nearest vehicle and in the rear of no. 0 intelligent vehicle in overtaking lane, and v_4 and a_4 are, respectively, for velocity and acceleration. We apply the perceptions of intelligent vehicle to represent the state space of each intelligent vehicle, the state set is s, $s = \{(l, v_0, d_1, v_1, d_2, v_2, d_3, v_3, d_4, v_4)\}$, where l indicates lane occupancy, values can take 1 or 2, respectively, take 1 means take up the carriageway, and 2 means occupied overtaking lane. State space dimensions are too high that will lead to information redundancy and not conducive to solving the problem. According to research literature [20] comprehensive consideration of vehicle states and the process of driving surroundings, take residual reaction time in the driving as state variables, the following calculation:

$$\begin{cases} t_1 = \dfrac{(d_1 - d_{m1})}{v_0}, \\ t_2 = \dfrac{(d_2 - d_{m2})}{v_2}, \\ t_3 = \dfrac{(d_3 - d_{m3})}{v_0}, \\ t_4 = \dfrac{(d_4 - d_{m4})}{v_4}, \end{cases} \tag{3}$$

where t_1 is the reaction time where in front of the vehicle decelerate in carriageway, d_{m1} is minimum safety distance in front of the carriageway which the intelligent vehicle with $-6m/s^2$ deceleration driving. t_2 is the reaction time where in rear of vehicle in carriageway, d_{m2} is minimum safety distance in rear of the carriageway which the intelligent vehicle with-$6m/s^2$ deceleration driving. t_3, d_{m3}, t_4, and d_{m4} are overtaking lane parameters. Then, the state space becomes $S = \{(l, t_1, t_2, t_3, t_4)\}$.

In order to avoid the decision of intelligent vehicle driving too frequently, when the intelligent vehicle between the two lanes is using the last moment of the decision, the independent decision of intelligent vehicle which is a set of MDPs actions: a_1 is velocity limit in carriageway, driving in the carriageway to velocity the task; a_2 is accelerated on lane, in the carriageway the velocity accelerate to the maximum safety velocity; a_3 is the minimum velocity limit driving in the carriageway; a_4 is on the overtaking lane to follow; a_5 is the task velocity on overtaking lane; a_6 maximum velocity on overtaking lane. The carriageway task velocity in the [60,100] km/h velocity interval is randomly generated, and max velocity 100 km velocity following is in front of vehicle no barrier-free vehicle where velocity to task velocity. If there is in a planned velocity v_{plane}, the calculation method [47] is as follows Table 1. The following distance is the shortest distance required for the intelligent vehicle to follow the vehicle ahead to stop (follow distance). For example, the intelligent vehicle running the carriageway, the $d_{follow} = (v_0^2/(-2a_0) + 5)$. Where $a_0 = -6m/s^2$ is the preset braking acceleration of vehicle deceleration, and 5m is the length of the vehicle.

Each ten seconds to update a task velocity. Overtaking velocity on overtaking lane is 110 km/h. In the process of independent driving of intelligent vehicles, the return rewards as follows:

$$r_{\text{safe}} = \begin{cases} -40 \text{ if collision}, \\ \min\ (t_1, t_2) \text{ if } l = 1 \text{ and } d_1 > 3 \text{ and } d_2 > 3, \\ \min\ (t_3, t_4) \text{if } l = 2 \text{ and } d_3 > 3 \text{ and } d_4 > 3, \\ -5 \text{ else}, \end{cases} \tag{4}$$

$$r_{\text{speed}} = \begin{cases} v_p - v_t \text{ if } v_p > v_t, \\ 0 \text{ else}. \end{cases}$$

$l = 1$ represents the distance between the intelligent vehicles which is greater than 3 meters in carriageway, in the same, $l = 2$ represents in overtaking lane, in the case of collision the value is -5. The MDP states interval is 10 seconds. When the vehicle driving distance among vehicles results in safety issues of mutual influence among vehicles, according to the shortest distance between the reaction time of the intelligent vehicle, and the intelligent vehicle perceived around context, then on the basis of context coordination among the behaviors of intelligent vehicles to collaborative intelligent vehicle actions.

3.3. Multivehicle Relationship Representation in the CG Model. When multi-intelligent vehicles coexist in the same environment and learn the policy at the same time, this scenario can be regarded as a MARL problem for multi-intelligent vehicles. Among them, the instability problem is caused by multi-intelligent vehicle learning in the same environment. However, teaching agents the coordination policies in this nonstationary environment is extremely challenging, especially when the intelligent vehicle still needs to deal with incomplete information caused by communication constraints or local observability constraints. The existing methods rarely focus on the coordination relationship between intelligent vehicles. In the CG [48] models in which the environment of multi-intelligent vehicles presents an undirected graph structure, the points in the graph represent

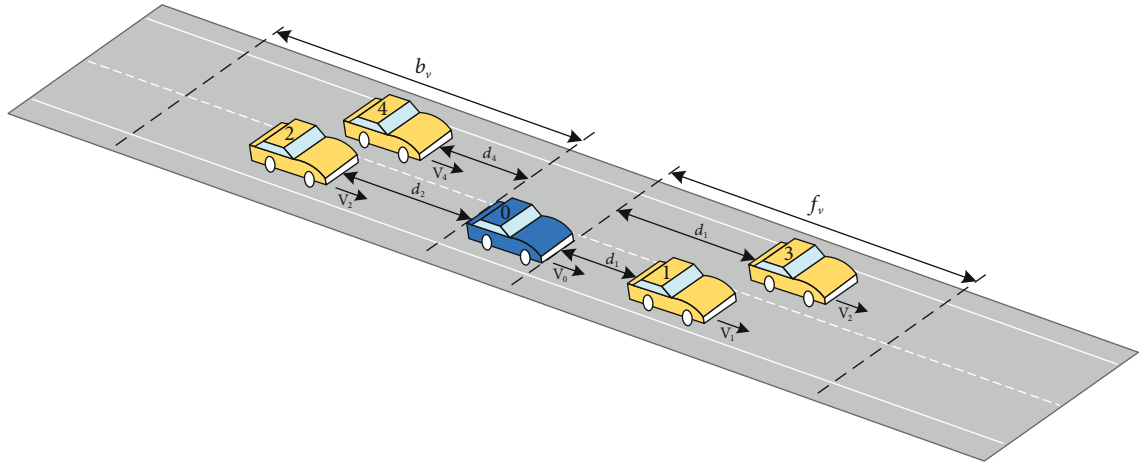

Figure 1: The state of vehicle.

Table 1: Planning velocity calculation method.

$d - d_{follow} > 10$	$v_1 - v_0 > 3.6$	$v_{plane} = v_0 + 0.25(d_1 - d_{follow}) + 1.5(v_1 - v_0)$
	$3.6 \geq v_1 - v_0 > -3.6$	$v_{plane} = v_0 + 0.25(d_1 - d_{follow}) + 1.0(v_1 - v_0)$
	$v_1 - v_0 \leq -3.6$	$v_{plane} = v_0 + 0.25(d_1 - d_{follow}) + 1.5(v_1 - v_0)$
$10 \geq d - d_{follow} > -4$	$v_1 - v_0 > 3.6$	$v_{plane} = v_0 + 0.5(d_1 - d_{follow}) + 1.5(v_1 - v_0)$
	$3.6 \geq v_1 - v_0 > -3.6$	$v_{plane} = v_0 + 0.5(d_1 - d_{follow}) + 1.5(v_1 - v_0)$
	$v_1 - v_0 \leq -3.6$	$v_{plane} = v_0 + 0.5(d_1 - d_{follow}) + 1.0(v_1 - v_0)$
$d - d_{follow} \leq -4$	$v_1 - v_0 > 3.6$	$v_{plane} = v_0 + 1.0(d_1 - d_{follow}) + 1.5(v_1 - v_0)$
	$3.6 \geq v_1 - v_0 > -3.6$	$v_{plane} = v_0 + 1.0(d_1 - d_{follow}) + 1.0(v_1 - v_0)$
	$v_1 - v_0 \leq -3.6$	$v_{plane} = v_0 + 1.0(d_1 - d_{follow}) + 1.5(v_1 - v_0)$

intelligent vehicles, while the edges represent the coordination relationship between agents. This setting provides a modeling basis and a theoretical basis for agents to achieve coordination decisions.

CGs are an effective method of solving the abovementioned problems. CGs can use a linear combination of local value functions to represent the global value function and subsequently reduce the influence of the number of intelligent vehicles on the complex computational domain. This approach of decomposition can be described using the undirected graph denoted by $G = (V, E)$ in which each node $i \in V$ represents an agent, and the edge $(i, j) \in E$ represents the corresponding agents that must make coordination decisions. On the basis of the CG model representing the coordination relationship of intelligent vehicles, the use of VE or maximum sum and other belief propagation algorithms [49] for solving the global maximum utility can be used to guide the vehicle to learn the coordination policies.

4. Method

First, we use the dynamic CG model to represent the objects that need to cooperate with a vehicle. Our dynamic coordination model uses DSF [50] as a danger relationship representation method of intelligent vehicles for dynamically constructing a CG that can represent the interaction relationship among the vehicles. On this basis, we can further refine the interaction weight by using the attention mechanism. Then, we use the graph convolution to simulate the belief propagation process to learn the driving policy. At the beginning of the training, we use the existing expert samples as a model to guide the policy learning, and we determine the potential coordination policies in the existing rules. After learning a policy under the guidance of the expert samples, we continue to explore new coordination policies. The relationship between models is shown in Figure 2.

4.1. Dynamic CG Generation Model Based on the Safety Field. We take the DSF model of automatic driving as the dynamic relationship generator of intelligent vehicles and express the interaction relationship between intelligent vehicles as a graph model. Through the DSF, the risk relationship between vehicles can be dynamically calculated to identify which intelligent vehicle needs to undergo cooperation. In using this method, the global policy learning problem can be simplified to a coordination policy learning problem among several small-scale intelligent vehicles, and

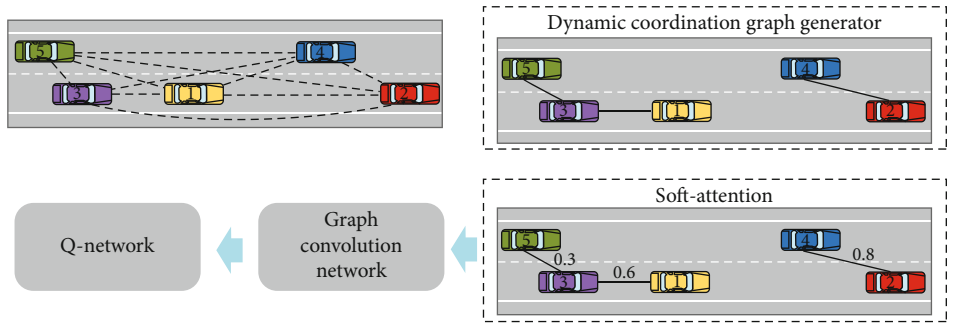

Figure 2: Structure of the multivehicle coordination learning graph convolutional RL based on dynamic CG.

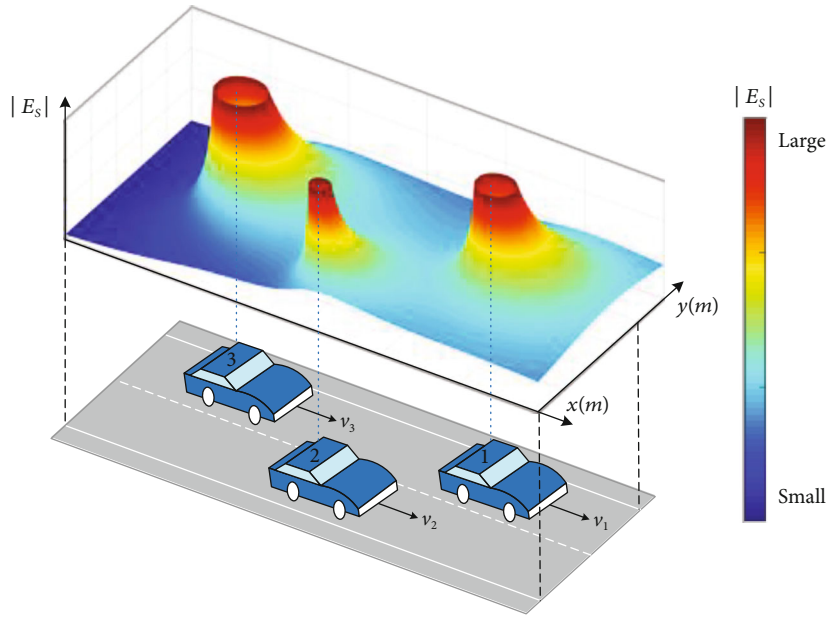

Figure 3: Driving safety field in highway scene.

the simple abstraction of the relationship between intelligent vehicles can be realized.

DSF is a kind of "physical field" characterizing the influence of various factors in the vehicle driving environment on the driving risk. As a physical quantity, DSF is calculated using the dynamic changes of various factors in the driving process. This study is aimed at investigating the scene of a two-lane highway where all of the vehicles are moving autonomous vehicles. As the vehicles are assumed to strictly abide by the traffic rules, we only consider the "kinetic energy field" and "behavior field" between vehicles. Figure 3 shows the field strength distribution of driving safety that can directly judge the degree of interaction between vehicles. We define the vertex set and edge set to construct the CG denoted by $G(V, E)$. The vertex set is composed of vehicle set $V = C$. Given a group of vehicles denoted by C, we check whether all vehicle pairs need to establish a coordination relationship according to the motion characteristics of the vehicles to avoid possible collision accidents.

A general scenario is used to illustrate in detail the analytic method of the coordination relationship between two vehicles. This scheme represents a general vehicle driving scenario on a highway. Among the vehicles, vehicles 1 and 2 are both running in the same lane (vehicle 1 is the leading vehicle, and vehicle 2 is the lagging vehicle). The corresponding speeds are v_1 and v_2, and the following distance is d. In this scenario, the field strength affecting the driving safety of vehicle 1 is composed of the kinetic energy field formed by vehicle 2 and the behavior field formed by its driving style. The direction of the field strength of these two fields at vehicle 1 is opposite the direction of v_1. According to the formula of the DSF model,

$$\left.\begin{aligned} E_{V\_21} &= \frac{GR_2M_2}{d^{k_1}} \exp(k_2 v_2) \\ E_{D\_21} &= E_{V\_21} D_{r2} \\ E_{S\_21} &= E_{V\_21} + E_{D\_21} \\ F_1 &= E_{S\_21} M_1 R_1 \exp[-k_2 v_1](1 + D_{r1}) \end{aligned}\right\}, \tag{5}$$

where E_{V_21} is the kinetic field strength of vehicle 1 received by vehicle 2, E_{D_21} is the behavior field strength of vehicle 1 received by vehicle 2, E_{S_21} is the total field strength of the

safety field received by vehicle 1, F_1 is the force received by vehicle 1, R_1 and R_2 are the road condition factors of vehicles 1 and 2, M_1 and M_2 are the masses of vehicles 1 and 2, and D_{r1} and D_{r2} are the risk factors of the driving style of vehicles 1 and 2, respectively. According to Equation (5), we can further derive

$$F_1 = \frac{GR_1R_2M_1M_2}{r^{k_1}}(1 + D_{r1})(1 + D_{r2}) \exp\left[-k_2(v_1 - v_2)\right], \tag{6}$$

where $G = 0.001$, $k_1 = 1$, $k_2 = 0.05$, $R_1 = R_2 = 1$, $M_1 = M_2 = 5000kg$, and $D_{r1} = D_{r2} = 0.2$. According to Equation (6), when the distance between the two vehicles decreases and the relative speed increases, the greater force between the two vehicles indicates the degree of driving danger. We set F_{safe} to 360 N [32]. If $F_1 > F_{\text{safe}}$; then, the driving risk between the two vehicles is great, and a coordination relationship between the two vehicles should be established. Through the DSF, we can then use the CG to express intelligent vehicles with interactive relationships.

4.2. Coordination Relationship Based on the Attention Mechanism. In the process of intelligent vehicle driving, each intelligent vehicle in the region should play a different role in the decision making. The method structure is shown in Figure 4. The weight of each edge in the CG should also be different. Therefore, we train an attention model to learn the importance weight of each edge in the CG. In this manner, multi-intelligent vehicles can be constructed as a complete graph structure, in which the intelligent vehicle is only connected with the intelligent vehicle that needs interaction. The weight on the edge describes the importance of each relationship. In our method, the dynamic CG is used to represent the interaction between two intelligent vehicles. The attention mechanism can calculate the importance of the interaction between vehicles and refine the relationship between the intelligent vehicles in the graph model. In the previous section, our dynamic graph model has used the DSF model to dynamically calculate and determine whether an interaction exists between any two intelligent vehicles as a means of preliminarily judging the relationship between agents. In this section, we use the attention mechanism to further determine the relationship weight.

The attention mechanism is a widely used technology for improving the accuracy of a model, and it can effectively learn the relationship representation between entities. We take each intelligent vehicle as an entity and use the multihead dot product attention as a convolution kernel, as this approach can effectively calculate the coordination relationship between vehicles. For each vehicle i, we calculate the relationship between this vehicle and its k-neighboring vehicles. The input features of each intelligent vehicle are mapped onto the query, key, and value representation of each independent attention head. For the attention head m, the relationship between vehicle i and neighbor vehicle j is calculated as follows:

$$\alpha_{ij} = \text{softmax}\left(e_{ij}\right) = \frac{\exp\left(e_{ij}\right)}{\sum_{k \in N_i} \exp(e_{ik})}, \tag{7}$$

where d_k is the dimension of the key (k) vector for preventing the dot product of two vectors from becoming too large. For each attention head, as shown in Equation (8), the value representations of all input features are weighted and aggregated by the learned relationships between vehicles.

$$h'_i = \sigma\left(\frac{1}{M}\sum_{m=1}^{M}\sum_{j \in N_i}\alpha_{ij}^{m}\mathrm{W}^{m}h_j\right). \tag{8}$$

The attention coefficient α_{ij} further refines the relationship between agents in the graph model, but the input order of the features is ignored by the kernel. In our proposed scheme, the multihead attention mechanism allows the kernel to simultaneously focus on the different relation representation subspaces by reusing the attention mechanism as a means of further stabilizing the training. The attention mechanism is used in this study to derive the weight of the coordination relationship (edge) between the intelligent vehicle (nodes) in the multivehicle CG.

4.3. Graph Convolutional Coordination RL. On the basis of the weighted CG model, we can learn the coordination policy. In traditional methods, belief propagation algorithms, such as VE or maximum sum, are used to solve the global utility problem, but the coordination function needs to be artificially defined in advance. The graph convolution method can function similarly to belief propagation on the graph by means of auto-learning, and it can aggregate messages from local to global to solve the joint utility [18]. In this study, we use the graph convolution method, an automatic learning belief propagation algorithm, to guide the policy learning of intelligent vehicles. In addition, the convolution kernel in the graph convolution network (GCN) [50] can further learn how to refine the relationship representation between agents and aggregate the contributions of neighboring agents with influences on the agents. GCN allows agents to adjust the focus according to the driving state of the vehicle, and it uses the superposition of multiple GCN layers to extract high-order relationship representations. The GCN can effectively capture the interaction between vehicles in a larger-scale domain to promote the coordination decision making among vehicles in a much larger range. For each intelligent vehicle, the generated state and relationship features are connected and inputted into the deep Q network. Then, the deep Q network selects the action to maximize the Q value and executes it through the exploration strategy. Each intelligent vehicle calculates the loss gradient through the global Q value and reward value and then applies the global loss gradient to all intelligent vehicles. This approach allows the intelligent vehicle to not only focus on maximizing its expected return but it can also consider how its decision will affect other intelligent vehicles. As such, the intelligent vehicle can learn the coordination policy. In addition, each intelligent vehicle is connected via

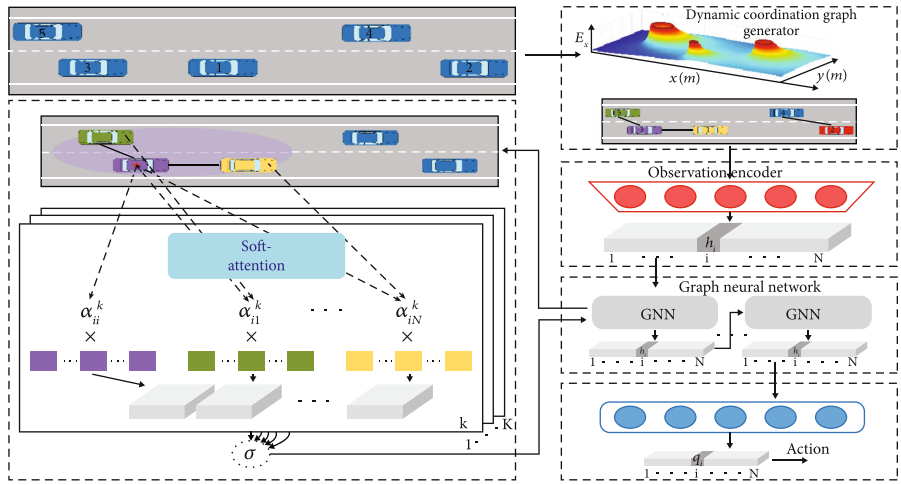

Figure 4: Graph convolutional reinforcement learning based on dynamic coordination graph.

the state code of nearby intelligent vehicles, which results in a much more stable environment from the perspective of single-intelligent vehicles. The forward reasoning can be formatted as follows:

$$\begin{aligned} \mathrm{h}_i &= \mathrm{Embed}\left(\mathrm{o}_i^t\right), \\ h_i^1 &= \mathrm{GCN}^1\left(\mathrm{h}_i\right), \\ &\cdots \\ h_i^L &= \mathrm{GCN}^L\left(\mathrm{h}_i^{L-1}\right), \\ Q\left(\mathrm{o}_i^t\right) &= \mathrm{Qnetwork}\left(h_i^L\right), \end{aligned} \tag{9}$$

where L is the number of GCN layers (each GCN represents a layer graph neural network structure), and $Q(\mathrm{o}_i^t)$ is the Q value of the final output. In the model for control-based graph convolution enhancement learning, the vehicles adopt the centralized training and distributed execution mode, and all vehicles share the weight. At each time step during the training, tuples (S, A, S', R, C) are stored in the experience playback buffer B. Then, we randomly take a small batch of s samples from B and minimize the Q loss as follows:

$$\mathrm{MSE}(\omega) = \frac{1}{S}\sum_{s}\frac{1}{N}\sum_{i=1}^{N}\left(y_i - Q(s_i, a_i, c_i; \omega)\right)^2, \tag{10}$$

where $y_i = r_i + \gamma \max_{a'} Q(o'_i, a'_i, c_i; \omega')s_i \in S$ is the current state of intelligent vehicle i, c_i is the adjacency matrix composed of intelligent vehicle and neighboring intelligent vehicles, γ is the discount factor (the model is parameterized by ω), and R is the immediate reward value of the intelligent vehicle. The Q loss gradients of all intelligent vehicles are accumulated, and the parameters are updated. As each intelligent vehicle only needs information from its k-neighboring intelligent vehicles during the execution of the action, the total number of intelligent vehicles can be ignored. This scheme allows the graph convolution RL method to be easily scaled and applied to large-scale multiagent systems, such as autonomous driving.

4.4. Model-Based Dynamic Graph Convolution RL. Although the existing DRL has good performance in many application scenarios, it continues to encounter serious learning efficiency problems when faced with complex tasks, especially sequential decision making. DRL often consumes numerous computing resources to achieve satisfactory results, which is far from the efficiency of humans. The blind trial of DRL in the early stage of learning greatly limits the learning efficiency of agents. The use of prior knowledge or experience to improve the algorithm performance is considered to be important for artificial intelligence. For example, imitation learning uses prior knowledge to directly guide each decision of RL, thus greatly speeding up the process of policy learning. To accelerate the early learning, we use the idea of model-based RL and reward shaping [51] to pretrain the model and introduce the expert samples generated by other excellent coordination algorithms as an additional reward value to guide the agent's decision making. In this manner, the learning efficiency of the model-free RL algorithm can be further improved.

Although some experimental data have shown that the method can significantly improve the learning efficiency of agents, implementing artificial expert rules as model constraints in complex environments is difficult. Especially in the MARL scenario, the coordination decision making between agents is hardly realized by establishing clear expert rules. Most prior knowledge in complex scenes is contained in rich expert samples, such as human driving data in traffic environments or driving data generated by other excellent algorithms. Therefore, we attempt to use an offline guidance method to guide the learning process of the model-free graph convolution RL by using the model constraints learned from the expert samples. In this manner, the intelligent vehicle can fully use the "existing knowledge," and it has a good learning effect in the early stage of training. In the later stage of learning, in the face of complex multiagent coordination tasks, we can realize exploratory learning

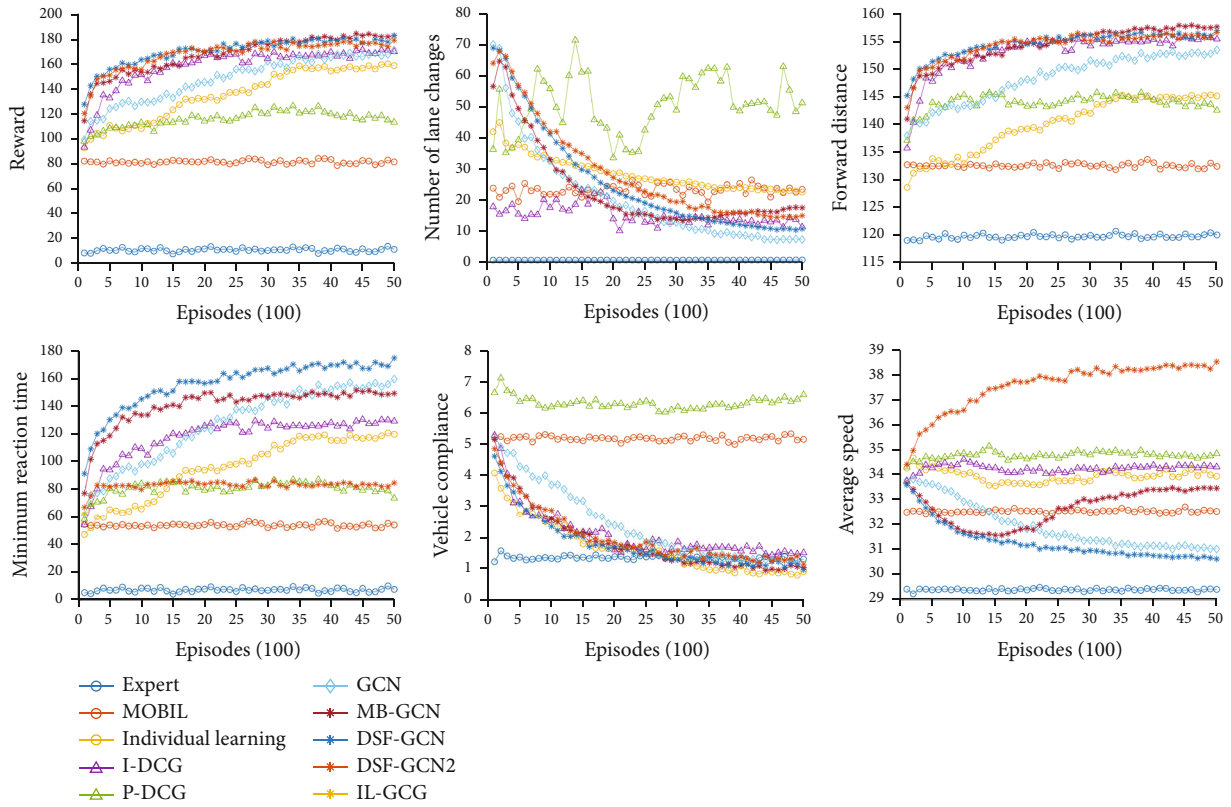

Figure 5: The average reward value and vehicle motion characteristics of various methods in 5-vehicle model.

through the continuous trial and error of the graph convolution RL, thus ensuring the high gradualness and generalization ability of the algorithm.

At the beginning of the training phase, we judge the similarity between the policy we have learned and the expert sample. If the result is consistent with the policy given by the expert sample; then, the reward value function $r_i = r(s_t, a_t) + r_d(s_t, a_t; \varphi)$ is adjusted, where $r(s_t, a_t)$ is reward under normal circumstances, and $r_d(s_t, a_t; \varphi)$ is additional rewards, which used to encourage the current policy to act like an expert. Then, the reward value function is fed back to the graph convolution RL and combined with the immediate reward value function of the environment feedback to derive the following formula:

$$\mathrm{MSE}(\omega) = \frac{1}{S}\sum_{s}\frac{1}{N}\sum_{i=1}^{N}\left(r_i + \gamma \max_{a'} Q\left(s_i', a_i', c_i; \omega'\right) - Q(s_i, a_i, c_i; \omega)\right)^2. \tag{11}$$

We extend this idea to the environment of multi-intelligent vehicles to guide the learning of intelligent vehicles by taking the excellent coordination driving sample data as the prior knowledge.

5. Experimental Results and Analysis

In the experimental environment of the highway, we used different methods to learn the driving policy of the vehicle. A total of 5000 rounds of training was utilized for all of the methods. Then, using the average of ten training results, we introduced the model-based (reward shaping) dynamic CG convolutional RL (MB-GCN) method guided by expert rules and the graph convolution RL based on the dynamic CG model of the driving security field (DSF-GCN). Finally, the graph convolutional RL (GCN) was evaluated. At the same time, we adjusted the linear ratio between the safety reward and the rapidity reward in the model to test the diversity of the developed driving policies. We defined the model that was trained by increasing the rapidity reward ratio as DSF-GCN2. To better explain the performance of the model, we used the classic mobility model of Mobil [52] and the expert rules [53]. Then, the two CG methods (I-DCG and P-DCG) [20] were compared with our method. Figure 5(a) shows the learning curve of the different methods with respect to the average rewards. MB-GCN, DSF-GCN, and DSF-GCN2 can finally converge to a higher average reward value and converge faster than all of the other models.

As expected, independent learning, mobile models, and expert rules do not consider the coordination relationship between agents; they may also reach the wrong driving decisions and perform poorly. Even though the mobile models and expert rules do not have the ability to relearn, the final result is much worse than those of the other methods. Unexpectedly, the P-DCG method had frequently selected the lane changing (fierce driving) decision because of its excessive pursuit of speed reward. Although this scenario had caused the vehicle to pursue a much higher driving speed, the driving safety of the vehicle was ignored. This finding

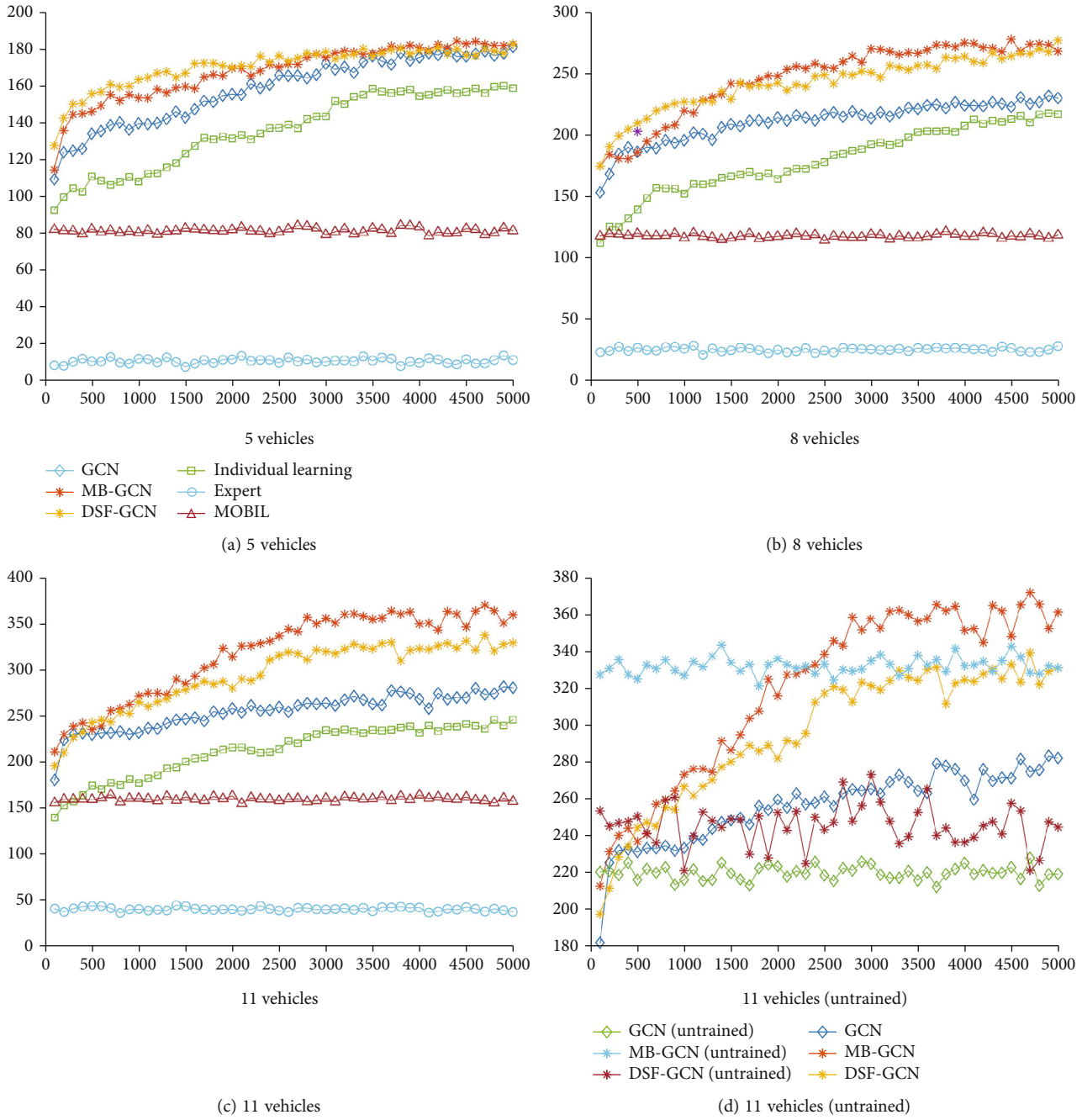

Figure 6: Average reward under different vehicle density.

is reflected in the factors, such as the shortest reaction time and the shortest forward distance.

We also compared the effects of the different reward ratios on the diversity of policy learning. Although the differences among DSF, GCN, and DSF-GCN2 are relatively small, the driving policies learned by DSF and GCN were more inclined to safe driving environments, which would quickly return to the driving lane on the premise of ensuring driving safety. DSF-GCN2, which had a more rapid reward, learned a more radical driving policy. Although this scheme could also ensure driving safety, its controlled autonomous driving vehicles tended to choose the advantages of speed and eventually formed a stable vehicle formation on the overtaking lane. This finding is a good simulation of real-life human drivers, as conservative drivers usually drive smoothly on driving lanes, and only in emergency situations will they choose to change to the overtaking lane. By contrast, radical drivers tend to continue occupying the overtaking lane to achieve the purpose of fast driving.

In the process of testing the performance of the different algorithms, we not only listed the curve of the final reward value but also a variety of indicators to evaluate the microattributes of the vehicles. The accumulated speed variation difference was included as an important index for evaluating

the driving comfort of the vehicles. The cumulative number of the lane changes of the vehicles appeared to be an important indicator for evaluating the accuracy of vehicle decision making. As shown in Figures 5(b)–5(f), a variety of model-based GCN methods have a better performance compared with the other methods in terms of the shortest response time, vehicle ride comfort, and number of lane changes. According to the policy that had been finally learned by the autonomous vehicle, we found that the MB-GCN method would eventually form a stable vehicle formation on the driving lane by coordinating the vehicle's lane-changing decision. More interestingly, the MB-GCN method would immediately change lanes to the overtaking lane when a vehicle was inserted in the rear, thus affecting the driving safety of the vehicle. After accelerating and driving a certain safe distance away from the vehicle behind it, the MB-GCN method would return to the safe driving lane.

However, the independent learning method has no coordination mechanism, and it could not learn the coordination policy. Moreover, due to the dynamic changes in the environment caused by the decisions of the surrounding vehicles, the scheme was often sensitive towards the selection of frequent lane-changing decisions. The relationship learning between intelligent vehicles can help intelligent vehicles to generate coordination policies, indicating that all methods in the GCN are better than the independent learning methods. However, the traditional GCN may rely too much on the reward in which an evaluation index brings to the vehicle and drops the policy learning into a local optimum. The DSF-GCN can be used to develop more complex driving policies, as it will hardly focus on conservative driving policies, but it will appropriately increase its driving speed while ensuring driving safety. This approach can greatly help to improve traffic efficiency.

To study the influence of vehicle density on the performance of the model, we conducted experiments in an autonomous vehicle environment by using different vehicle densities. As shown in Figures 6(a)–6(c), as the density of the vehicles increases, the differences between the various methods become more apparent. Among them, MB-GCN remains to be the method that can obtain the highest reward value. This finding fully proves the benefits of our method in terms of learning efficiency and learning effect. The increase in the number of vehicles had caused an increase in environmental instability, which subsequently caused great obstacles to the GCN for learning the relationship between agents. A good convergence effect can be achieved in a low-density five-car environment. However, with the increase in the number of agents, although a quick convergence can be attained, the final learning results indicate that the values can prematurely fall into the local optimal solution. Especially during traffic congestion, the learning of this policy can hardly solve the coordinated decision making among the multiple vehicles. However, when the traffic density is low, the simple relationship between vehicles is beneficial to the learning of the GCN.

In Figure 6(d), we show the results of using the previously trained model parameters directly in the 11 car environment without retraining. It is worth noting that the MB-GCN method without retraining can still get the highest reward value, and the gap with the retraining method is the smallest, which fully proves the scalability of MB-GCN. Interestingly, the reward value of all retrained GCN methods is slightly higher, in which the vehicle speed is reduced, and the number of lane changes is also significantly reduced, but the safety of vehicles is not greatly affected. The reason is that in the case of low density, due to the small number of vehicles, the possibility of collision between vehicles is small. The learning of vehicle driving decision mainly focuses on how to accelerate through the overtaking lane, so as to get rid of traffic congestion quickly and get a better driving environment for vehicles. However, with the increase of vehicle density, vehicles need to walk longer to get a better driving environment, and the increase of collision probability makes vehicles tend to choose more conservative driving decisions, so the driving decisions learned tend to avoid collision accidents.

6. Conclusion

We focus on promoting the coordination among multi-intelligent vehicles through the relationship learning of vehicles and propose a dynamic CG convolutional RL method that introduces model constraints. By combining the method of constructing a dynamic CG with the soft attention mechanism, the interference of irrelevant vehicles can be effectively removed, and the learning efficiency of the algorithm can be accelerated while ensuring better progressive performance. The vehicle can adapt to the dynamic changes of the underlying graph, and it can use the potential features of the relational kernel convolution to learn coordination policies from the gradually increasing receptive field. The method of intensive training allows the gradient of an intelligent vehicle to not only counteract itself but also counteract other intelligent vehicles in its receptive domain. In this manner, the intelligent vehicles can learn to be coordinated. At the same time, excellent driving samples have been used as the training data to combine the model guided by the reward value with the graph convolution RL without the model. This approach can reduce the invalid exploration of intelligent vehicles and guide them to ignore certain driving policies resulting from the reduction of their own speed in view of obtaining the driving policy that balances safety and efficiency. In the scene of multivehicle cooperative driving on highways, the convolution RL of the dynamic cooperative graph is significantly better than those of the existing methods.

In the future, we will continue the research on multivehicle cooperative driving. At present, our research focuses on fully cooperative automatic driving. In the next step, we will study man-machine hybrid multivehicle cooperative driving. In the man-machine hybrid cooperative driving mode, it is difficult for an autonomous vehicle to drive safely and efficiently with a human driver.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. U1808206.

References

[1] B. R. Kiran, I. Sobh, V. Talpaert et al., "Deep reinforcement learning for autonomous driving: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2021.

[2] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of Field Robotics*, vol. 37, no. 3, pp. 362–386, 2020.

[3] T. Shi, P. Wang, X. Cheng, C.-Y. Chan, and D. Huang, "Driving decision and control for automated lane change behavior based on deep reinforcement learning," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 2895–2900, Auckland, New Zealand, 2019.

[4] Z. Zhu and H. Zhao, "A survey of deep rl and il for autonomous driving policy learning," in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–23, 2021.

[5] R. Abduljabbar, H. Dia, S. Liyanage, and S. A. Bagloee, "Applications of artificial intelligence in transport: An overview," *Sustainability*, vol. 11, no. 1, p. 189, 2019.

[6] D. Ye, Z. Liu, M. Sun et al., "Mastering complex control in moba games with deep reinforcement learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 4, pp. 6672–6679, 2020.

[7] S. Wang, J. Duan, D. Shi et al., "A data-driven multi-agent autonomous voltage control framework using deep reinforcement learning," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4644–4654, 2020.

[8] J. Chen, S. E. Li, and M. Tomizuka, "Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5068–5078, 2022.

[9] M. Zhu, Y. Wang, Z. Pu, J. Hu, X. Wang, and R. Ke, "Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving," *Transportation Research Part C: Emerging Technologies*, vol. 117, article 102662, 2020.

[10] Y. Zhang, P. Sun, Y. Yin, L. Lin, and X. Wang, "Human-like autonomous vehicle speed control by deep reinforcement learning with double Q-learning," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, China, 2018IEEE.

[11] L. Chen, X. Hu, B. Tang, and Y. Cheng, "Conditional DQN-based motion planning with fuzzy logic for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 2966–2977, 2022.

[12] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, *CARLA: An Open Urban Driving Simulator*, PMLR, 2017.

[13] G. Habibi and J. P. How, "Human trajectory prediction using similarity-based multi-model fusion," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 715–722, 2021.

[14] S. Li, Y. Wu, X. Cui, H. Dong, F. Fang, and S. Russell, "Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 4213–4220, 2019.

[15] D. Cao, J. Zhao, W. Hu et al., "Data-driven multi-agent deep reinforcement learning for distribution system decentralized voltage control with high penetration of PVs," *IEEE Transactions on Smart Grid*, vol. 12, no. 5, pp. 4137–4150, 2021.

[16] Y. Liu, W. Wang, Y. Hu, J. Hao, X. Chen, and Y. Gao, "Multi-agent game abstraction via graph attention neural network," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 5, pp. 7211–7218, 2020.

[17] W. Böhmer, V. Kurin, and S. Whiteson, *Deep coordination graphs*, PMLR, 2020.

[18] S. Li, J. K. Gupta, P. Morales, R. Allen, and M. J. Kochenderfer, "Deep implicit coordination graphs for multi-agent reinforcement learning," in *Proceedings of the 20th International Conference on Autonomous Agents and Multi Agent Systems*, pp. 764–772, United Kingdom, 2021.

[19] J. Wang, M. Xu, L. Jiang, and Y. Song, "Attention-based deep reinforcement learning for virtual cinematography of 360° videos," *IEEE Transactions on Multimedia*, vol. 23, pp. 3227–3238, 2021.

[20] C. Yu, X. Wang, X. Xu et al., "Distributed multiagent coordinated learning for autonomous driving in highways based on dynamic coordination graphs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 735–748, 2020.

[21] D. Tokody, I. J. Mezei, and G. Schuster, "An overview of autonomous intelligent vehicle systems," in *Vehicle and Automotive Engineering*, pp. 287–307, Springer, 2017.

[22] D. Loiacono, A. Prete, P. L. Lanzi, and L. Cardamone, "Learning to overtake in TORCS using simple reinforcement learning," in *IEEE Congress on Evolutionary Computation*, Barcelona, Spain, 2010IEEE.

[23] F. Guo and Z. Wu, "A deep reinforcement learning approach for autonomous car racing," in *International Conference on E-Learning and Games*, Cham, 2018Springer.

[24] J. Wang, Q. Zhang, D. Zhao, and Y. Chen, "Lane change decision-making through deep reinforcement learning with rule-based constraints," in *2019 International Joint Conference on Neural Networks (IJCNN)*, Budapest, Hungary, 2019IEEE.

[25] V. Talpaert, I. Sobh, B. R. Kiran et al., "Exploring applications of deep reinforcement learning for real-world autonomous driving systems," 2019, http://arxiv.org/abs/1901.01536.

[26] H. Chae, C. M. Kang, B. Kim, J. Kim, C. C. Chung, and J. W. Choi, "Autonomous braking system via deep reinforcement learning," in *2017 IEEE 20th International conference on intelligent transportation systems (ITSC)*, Yokohama, Japan, 2017IEEE.

[27] F. Belletti, D. Haziza, G. Gomes, and A. M. Bayen, "Expert level control of ramp metering based on multi-task deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1198–1207, 2018.

[28] K. Makantasis, M. Kontorinaki, and I. Nikolos, "Deep reinforcement-learning-based driving policy for autonomous road vehicles," *IET Intelligent Transport Systems*, vol. 14, no. 1, pp. 13–24, 2020.

[29] J. Dong, S. Chen, Y. Li et al., "Spatio-weighted information fusion and DRL-based control for connected autonomous vehicles," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, Rhodes, Greece, 2020IEEE.

[30] Y. Chen, C. Dong, P. Palanisamy, P. Mudalige, K. Muelling, and J. M. Dolan, "Attention-based hierarchical deep reinforcement learning for lane change behaviors in autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, California, 2019.

[31] A. Folkers, M. Rick, and C. Büskens, "Controlling an autonomous vehicle with deep reinforcement learning," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, 2019IEEE.

[32] D. E. Moriarty and P. Langley, "Learning cooperative lane selection strategies for highways," *AAAI/IAAI*, vol. 1998, pp. 684–691, 1998.

[33] A. G. Cunningham, E. Galceran, D. Mehta, G. Ferrer, R. M. Eustice, and E. Olson, "MPDM: multi-policy decision-making from autonomous driving to social robot navigation," *Control Strategies for Advanced Driver Assistance Systems and Autonomous Driving Functions*, vol. 476, pp. 201–223, 2019.

[34] C. Fei, B. Wang, Y. Zhuang et al., "Triple-GAIL: a multi-modal imitation learning framework with generative adversarial nets," in *In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pp. 2929–2935, Yokohama, Japan, 2021.

[35] O. Sharma, N. C. Sahoo, and N. B. Puhan, "Recent advances in motion and behavior planning techniques for software architecture of autonomous vehicles: a state-of-the-art survey," *Engineering Applications of Artificial Intelligence*, vol. 101, article 104211, 2021.

[36] S. Han and F. Miao, "Behavior planning for connected autonomous vehicles using feedback deep reinforcement learning," 2020, http://arxiv.org/abs/2003.04371.

[37] A. Alizadeh, M. Moghadam, Y. Bicer, N. K. Ure, U. Yavas, and C. Kurtulus, "Automated lane change decision making using deep reinforcement learning in dynamic and uncertain highway environment," in *2019 IEEE intelligent transportation systems conference (ITSC)*, Auckland, New Zealand, 2019IEEE.

[38] P. Wang, C. Chan, and A. de La Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, China, 2018IEEE.

[39] W. Yuan, M. Yang, Y. He, C. Wang, and B. Wang, "Multi-reward architecture based reinforcement learning for highway driving policies," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, 2019IEEE.

[40] L. Wang, F. Ye, Y. Wang et al., "A Q-learning foresighted approach to ego-efficient lane changes of connected and automated vehicles on freeways," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, 2019IEEE.

[41] H. Bey, F. Dierkes, S. Bayerl, A. Lange, D. Faßender, and J. Thielecke, "Optimization-based tactical behavior planning for autonomous freeway driving in favor of the traffic flow," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, 2019IEEE.

[42] M. Sefati, J. Chandiramani, K. Kreisköther, A. Kampker, and S. Baldi, "Towards tactical behaviour planning under uncertainties for automated vehicles in urban scenarios," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–7, IEEE, Yokohama, Japan, 2017.

[43] J. Jiang, C. Dun, T. Huang, and Z. Lu, "Graph convolutional reinforcement learning," in *In International Conference on Learning Representations*, New Orleans, LA, USA, 2019, September.

[44] H. Si, G. Tan, Y. Peng, and J. Li, "Dynamic coordination-based reinforcement learning for driving policy," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 6836778, 18 pages, 2022.

[45] S. D. Whitehead and L. Lin, "Reinforcement learning of non-Markov decision processes," *Artificial Intelligence*, vol. 73, no. 1-2, pp. 271–306, 1995.

[46] S. Carta, A. Ferreira, A. S. Podda, D. Reforgiato Recupero, and A. Sanna, "Multi-DQN: an ensemble of deep Q-learning agents for stock market forecasting," *Expert Systems with Applications*, vol. 164, article 113820, 2021.

[47] X. Li, X. Xu, and L. Zuo, "Reinforcement learning based overtaking decision-making for highway autonomous driving," in *Sixth international conference on intelligent control and information processing*, pp. 336–342, IEEE, Wuhan, China, 2015.

[48] J. R. Kok, M. T. Spaan, and N. Vlassis, "Multi-robot decision making using coordination graphs," *Proceedings of the 11th International Conference on Advanced Robotics, ICAR*, vol. 3, 2003.

[49] J. R. Kok and N. Vlassis, "Collaborative multiagent reinforcement learning by payoff propagation," *Journal of Machine Learning Research*, vol. 7, pp. 1789–1828, 2006.

[50] J. Wang, J. Wu, and Y. Li, "The driving safety field based on driver–vehicle–road interactions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2203–2214, 2015.

[51] Y. Dong, X. Tang, and Y. Yuan, "Principled reward shaping for reinforcement learning via Lyapunov stability theory," *Neurocomputing*, vol. 393, pp. 83–90, 2020.

[52] J. Harri, F. Filali, and C. Bonnet, "Mobility models for vehicular ad hoc networks: a survey and taxonomy," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 4, pp. 19–41, 2009.

[53] N. C. Basjaruddin, K. Kuspriyanto, D. Saefudin, E. Rakhman, and A. M. Ramadlan, "Overtaking assistant system based on fuzzy logic," *Telkomnika*, vol. 13, no. 1, p. 76, 2015.