



Research Article

A Resource Allocation Scheme for Intelligent Tasks in Vehicular Networks

Jiujiu Chen ^{1,2}, Caili Guo,^{1,2} Chunyan Feng,^{1,2} Chuanhong Liu,¹ Xin Sun,¹ and Jun Liu ³

¹Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China

²Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China

³Tsinghua University, Beijing 100080, China

Correspondence should be addressed to Jiujiu Chen; chenjiujiu@bupt.edu.cn

Received 25 May 2022; Accepted 25 August 2022; Published 27 September 2022

Academic Editor: Lisheng Fan

Copyright © 2022 Jiujiu Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Lots of resource-consuming intelligent tasks need to be handled in vehicular networks, and traditional resource allocation schemes are hard to meet the intelligent demands. Therefore, this paper proposes a task-oriented resource allocation scheme for intelligent tasks in vehicular networks. First, we propose a task-oriented communication system and formulate a resource allocation problem, which is aimed at maximizing the task performance. Second, based on the system model, an intelligent task-oriented resource allocation optimization criterion is proposed, which is formulated as a mathematical model, and its parameters are solved by the proposed gradient descent-based algorithm. Third, to solve resource allocation problem, a multiagent deep Q-network-(MADQN-) based algorithm is proposed, whose convergence and complexity are further analyzed. Last, experiments on real datasets verify the performance advantages of our proposed algorithms.

1. Introduction

Recently, more and more vehicles are equipped with high-definition cameras to enhance visual perception [1]. More than 90% of the driving environment information can be collected and acquired by the cameras [2]. At the same time, the vehicles use artificial intelligence technology to fully analyze the large amount of data collected by the vehicular cameras, so as to complete the various tasks in the process of driving [3]. These intelligent tasks, based on a deep neural network, such as classification, detection, and recognition, put forward a huge demand for computing resource at the vehicle ends.

Thanks to the development of Internet of vehicles (IoV) technology in recent years [4], a feasible solution is that the vehicles transmit data to the edge server to complete the intelligent tasks. Then, the edge server feeds back the calculation results to the vehicles, so as to support the intelligent needs of various applications, such as assisted driving and automatic driving in IoV scenarios.

However, a large amount of multimedia data transmission and computing task offloading from the vehicles to the edge server bring great pressure to the communication resources. Therefore, in order to promote the integration of communication and computing processes in the IoV system, it is urgent to study efficient resource allocation schemes to improve the resource utilization rate in the vehicular networks and better serve the intelligent tasks of the vehicles.

1.1. Related Works and Motivations. Existing studies on resource allocation in the IoV are mainly oriented to network efficiency or user experience, with the purpose of maximizing quality of service (QoS) or quality of experience (QoE). Vehicle mobility and service diversity in the IoV scenarios lead to different QoS requirements, so QoS-based resource allocation schemes focus on how to build QoS models suitable for IoV scenarios [5, 6] and how to design resource allocation algorithms based on QoS models, such as channel selection [7], power control [8], and spectrum

sharing [9]. Although these QoS-based schemes can improve the traditional performance metrics such as capacity, they do not handle the subjective needs.

Compared to network efficiency, user experience-oriented demands are more subjective. QoE-based resource allocation schemes focus more on the needs of human users. Most works focus on QoE modeling to meet the diverse needs of humankind users [10]. For example, safety traffic services have higher requirements on video definition and resolution, while entertainment services have higher requirements on video smoothness. Moreover, some works pay attention to algorithm designing for resource allocation to meet QoE requirements in the IoV scenarios. The main method is to map QoE requirements to communication resource requirements such as spectrum and power [11, 12]. And the other is to establish the mapping or association between QoE and QoS and then use the multiobjective resource allocation methods to improve the quality of user experience [13, 14].

Most of the existing resource allocation schemes for the IoV scenarios mentioned above ignore the needs for intelligent tasks and seldom consider the contents or semantics of data. When the transmitted data is used for intelligent tasks such as classification, detection, and recognition, its goal is no longer network efficiency or user experience, but the understanding or analysis accuracy of visual contents or semantics, so the traditional QoS or QoE-based resource allocation schemes are no longer optimal [15]. A more suitable resource allocation scheme is needed for intelligent task-oriented communication system.

It is worth noting that the future network is becoming more and more intelligent. It is no longer simply concerned about the pursuit of transmission speed but pays more attention to the demands of intelligence [16]. The intelligent requirements of 6G make the research shift from the traditional communication based on Shannon's framework to the semantic or goal-oriented communication [17]. Recently, some works focus on intelligent end-to-end semantic communication systems based on deep learning [18, 19]. They propose that image or video features, instead of the full data, are uploaded to the servers for data analyses. Although transmitting features can save wireless resources, they are not suitable for all kinds of tasks since the full data is required to be archived in the server for future investigations. Moreover, even if they have achieved good results in solving the end-to-end intelligent tasks' requirements, they ignore the problems of resource limitation. Both sending and receiving ends require a lot of computing power, and it is difficult to support the implementation of those algorithms under the condition of resource shortage.

In the IoV scenarios where intelligence and network connectivity are highly integrated, it is urgent to explore the balance between network intelligence and efficient resource utilization. According to the above analysis, the main challenges are as follows:

- (i) How to design a multimedia data transmission system for vehicles with limited computing capacity, to

improve transmission efficiency and meet the needs of intelligent computing?

- (ii) How to design an optimization criteria for resource allocation in intelligent scenarios of the IoV, to solve the contradiction between intelligent task requirements and traditional resource allocation methods?
- (iii) How to design a resource allocation algorithm with high stability and low complexity to adapt to the dynamic changing environment of the IoV?

1.2. Contributions and Organization. To address these challenges, in our previous work [20, 21], we made preliminary exploration and proposed a single-task oriented spectrum allocation algorithm. In order to meet the requirements of multi-tasks and to extend the flexibility of resource allocation scheme in mobile scenarios, this paper further proposes a deep reinforcement learning-based resource allocation scheme for multiple intelligent tasks in vehicular networks, which are aimed at maximizing the performance of intelligent tasks under resource constraints. Contributions of this paper are as follows:

- (i) We design a task-oriented communication system for multi-intelligent tasks in the IoV scenario. Based on the proposed system model, we construct a multivariable resource allocation optimization problem, which is aimed at maximizing the performance of intelligent tasks
- (ii) We propose an intelligent task-oriented resource allocation optimization criterion, which is expressed as a mathematical model, and we design a gradient descent-based algorithm for solving model parameters
- (iii) We propose a multiagent deep Q-network-(MADQN-) based algorithm to solve the resource allocation problem, and analyze the convergence and complexity of the proposed algorithm

In addition, we verify the performance advantages of the proposed algorithms based on the new dataset found by us and the existing datasets.

The rest of this paper is organized as follows. In Section 2, the system model is described and the resource allocation problem is formulated. In Section 3, a detailed description about resource allocation optimization criteria is given. In Section 4, a MADQN-based resource allocation algorithm is given. The numerical results are shown in Section 5, followed by the concluding remarks in Section 6.

2. System Model and Problem Formulation

2.1. Scenario and System Model. A typical scenario of task-oriented vehicular network is shown in Figure 1. The network consists of an edge server and multiple vehicles equipped with cameras. Vehicles are randomly distributed in the server's coverage area. Vehicles perform data collecting and preprocessing, and an edge server completes data storage and intelligent tasks, such as classification, detection, and reidentification (Re-ID).

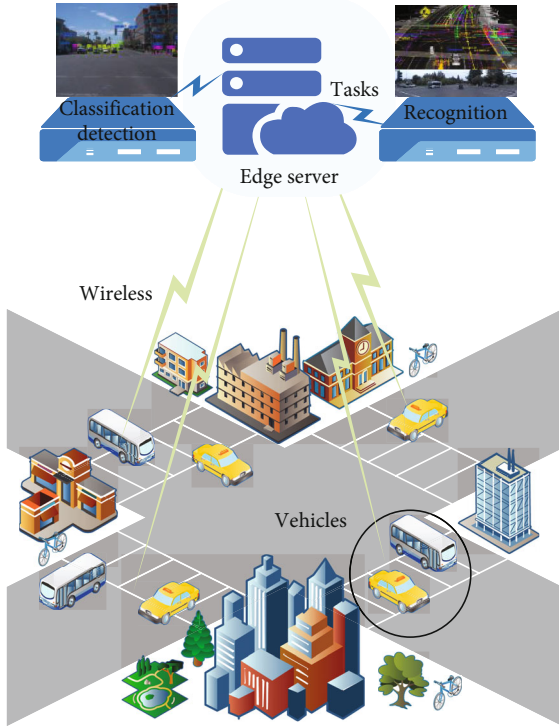


FIGURE 1: The task-oriented vehicular network scenario.

Specifically, vehicles perceive the surrounding environment through cameras, including traffic signals and moving objects (such as pedestrians and other vehicles). This visual environmental information, like image and video data perceived by vehicles, needs to be further analyzed. Considering the huge amount of collected multimedia data and the limited computing capacity of the vehicles, the data is transmitted to the edge server through wireless channels to complete the data storage, data analysis, and subsequent intelligent tasks.

Therefore, the goal of the communication system is to improve the tasks' performance as much as possible under the condition of limited resources. To achieve that goal, we design a task-oriented system, as shown in Figure 2.

The proposed system is divided into three parts: vehicles, wireless channel, and edge server. The vehicles contain three modules: (1) collecting module, which is used to collect original multimedia data and complete data preprocessing; (2) encoding module, completing the source coding and digital signal modulation; and (3) control module, which controls the transmission power and selects transmission frequency band and channel. A wireless channel is a mobile time-varying fading channel. The edge server consists of two modules: (1) decoding module, which completes the demodulation and decoding of signals to restore the original data, and (2) computing module, including data storage (for data analysis and model training), intelligent tasks performed by neural networks, optimization criteria modeling, and system resource allocation.

In this system, the vehicles select appropriate encoding data rate, power, and bandwidth according to the results of resource allocation and transmit the compressed data to

the edge server through wireless channel. The edge server restores the data and then completes the tasks. The main purpose of the system is to achieve the best task performance by reasonable resource allocation, which is a multivariable resource allocation optimization problem.

2.2. Resource Allocation Problem Formulation. Due to the proposed communication system being task-oriented, the communication goal is to maximize the task performance. Based on the proposed system, task performance is mainly related to the quality of received multimedia data, which is mainly affected by compression rate and bit error rate. Hence, the resource optimization criteria can be formulated as

$$F_m^{\text{task}}(q_m, p_m) = \lambda_m \cdot F_m^{\text{task}}(q_m) \cdot F_m^{\text{task}}(p_m), \quad (1)$$

where $m \in \{1, 2, \dots, M\}$ is the vehicle index, M is the maximum number of vehicles, F_m^{task} denotes the task performance such as accuracy or mAP, λ_m is the weight coefficient, and q_m and p_m are the compression rate and bit error rate, respectively.

Here, F_m^{task} denotes the task performance like accuracy and mAP, which is the metric of resource allocation schemes for intelligent tasks. From Equation (1), the task performance values are related to traditional communication performance metrics, compression rate, and bit error rate. The following is the explanation. According to Figure 2, the communication progress involves two processes, encoding and transmission, in which compression and noise are the key factors leading to data quality decline and affecting the performance of subsequent tasks. Therefore, Equation (1) describes the relationship between the performance of intelligent tasks and communication parameters. It reveals that task performance like accuracy and mAP can be improved by reasonable communication resource allocation. Besides, the accurate mathematical model of Equation (1) is the basis for resource allocation problems.

Accordingly, the key of optimization problem formulation lies in the calculation of compression rate and bit error rate.

2.2.1. Compression Rate. According to [22], the compression rate is related to the source coding scheme, data block size, and encoder packet number; hence, it can be expressed as

$$q_m = 1 - \frac{L_m \cdot G_m}{S_m}, \quad (2)$$

where L_m is the encoder packet length, G_m is the packet size, and S_m is the block size. Then, the encoded data rate is

$$R_m^B = \frac{L_m \cdot G_m}{T_m^G}, \quad (3)$$

where R_m^B is the data rate and T_m^G is the block duration.

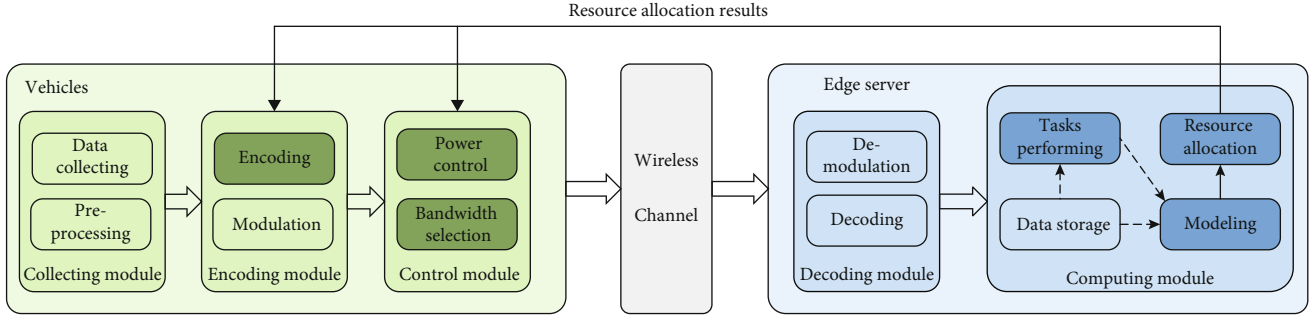


FIGURE 2: The structure of our proposed task-oriented communication system in a vehicular network.

2.2.2. *Bit Error Rate.* Based on [23], with quadrature amplitude modulation (QAM), the bit error rate is expressed as

$$p_m = \frac{2(1 - (1/\sqrt{N}))}{\log_2(\sqrt{N})} Q\left(\sqrt{\left(\frac{3\log_2(\sqrt{N})}{N-1}\right) 2E\left(\frac{P_m h_m}{n_0 R_m}\right)}\right), \quad (4)$$

where N is the modulation order. $Q(\cdot)$ is the Q-function, and $Q(x) = \int_x^{+\infty} (1/\sqrt{2N})e^{-t^2/2} dt$. $E(\cdot)$ is the expectation function. P_m is the transmission power, h_m is the channel gain, n_0 is the noise power spectral density, and $R_m = B_m \log_2(1 + (P_m h_m/n_0 B_m))$ is the transmission rate, where B_m is the bandwidth, P_m is the power, and h_m is the channel gain of each vehicle.

The channel gain h_m is modeled as an independent random variable, accounting for both large-scale fading h_m^L (contains path loss h_{pl} and shadowing h_{sd}) and small-scale fading effects h_m^S . Since the large-scale fading of channels is typically determined by vehicle locations, which do not change too much during transmission slots [24]. Here, the path loss is modeled as $h_{pl} = 148.1 + 37.6\log_{10}(d_m)$ (dB), where d_m (in km) is the distance between the m -th vehicle and the edge server. Shadowing is modeled by using a log-normal distribution, with a standard deviation of 8 dB and zero mean [24]. However, the small-scale fading components might change. Considering the dynamic nature of the small-scale fading, we model the time-varying coefficients as independent first-order auto-regressive processes [25], given by

$$h_m^S(t) = \rho_m(t_e)h_m^S(t - t_e) + e_h, \quad (5)$$

where t_e is the time interval, e_h is the process noise sequence from a $\mathcal{CN}(0, 1 - \rho_m^2(t_e))$ distribution, $\rho_m(t_e)$ is the channel autocorrelation function, and $\rho_m(t_e) = J_0(2\pi c v_m t_e / f_c)$, where $J_0(\cdot)$ is the zero-order Bessel function of the first kind, c is the velocity of light, f_c is the band mid-frequency, and v_m is the velocity of the m -th vehicle.

2.2.3. *Optimization Problem.* The goal of the proposed system is to transmit images and videos to the edge server, the server returns results in time. Therefore, the optimization

problem is to allocate resources to transmit images or videos under constraints to achieve the best task performance, which can be formulated as

$$\begin{aligned} \text{P1: } & \max_{L_m, B_m, P_m} \sum_{m=1}^M F_m^{\text{task}}(q_m, p_m) \\ \text{s.t. } & \text{C1: } \sum_{m=1}^M \frac{L_m \cdot G_m}{T_m^G} \leq R_{\max}^B \\ & \text{C2: } \sum_{m=1}^M B_m \leq B_{\max} \\ & \text{C3: } \sum_{m=1}^M P_m \leq P_{\max} \\ & \text{C4: } \frac{L_m \cdot G_m}{R_m(B_m, P_m)} \leq \tau \\ & \text{C5: } F_m^{\text{task}}(q_m) \geq f_{\min} \\ & \text{C6: } F_m^{\text{task}}(p_m) \geq f_{\min} \\ & \text{C7: } q_m(L_m) \leq q_e \\ & \text{C8: } p_m(B_m, P_m) \leq p_e \end{aligned} \quad (6)$$

where R_{\max}^B , B_{\max} , and P_{\max} denote the total data rate, power, and bandwidth of communication system, respectively. τ denotes the maximum transmission delay allowed by the system. f_{\min} , q_e , and p_e denote the task performance threshold, compression rate threshold and bit error rate threshold, respectively. The constraints are described in detail below.

C1 ~ C3 represents the constraints of resources, which means the sum of data rate, bandwidth, and power allocated to each vehicle are not larger than the total available data rate resource R_{\max}^B , the total available bandwidth B_{\max} , and the total available power P_{\max} , respectively. C4 represents the constraint of delay requirement, which means the transmission time must be shorter than the maximum delay τ allowed by the system. C5 ~ C6 represents the constraints of task performance requirements, in order to ensure that the results of intelligent tasks can meet the minimum requirements f_{\min} of vehicle users. C7 ~ C8 represents the

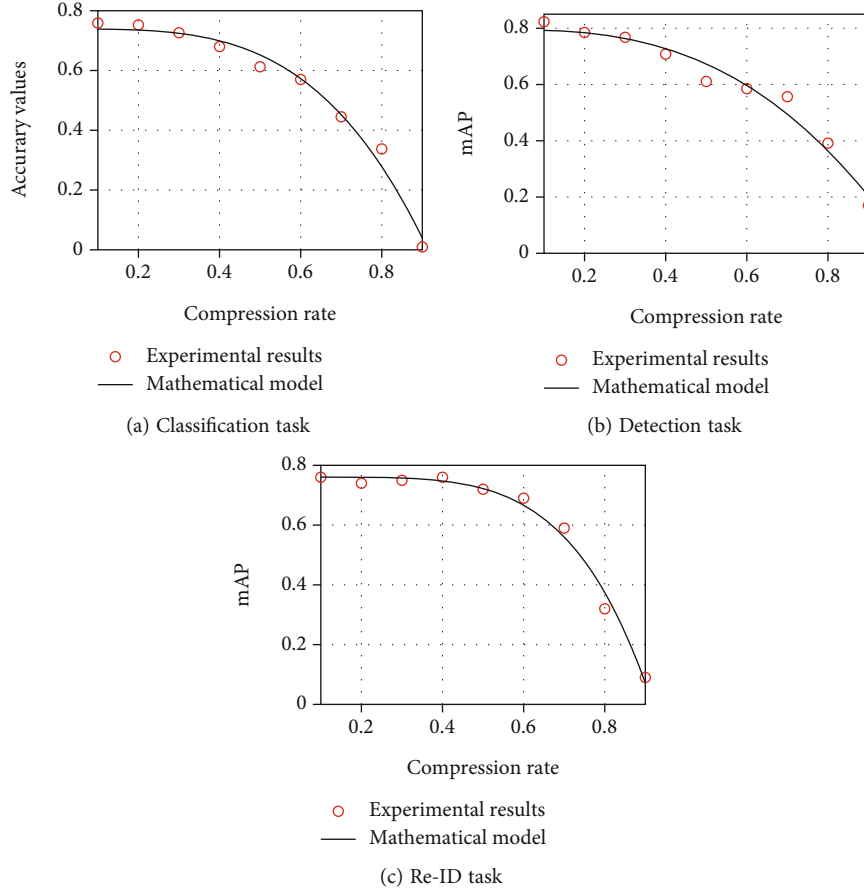


FIGURE 3: The relationship between task performance and compression rate with different tasks.

constraints of communication requirements. C7 represents the constraint of the compression rate, which guarantees the image and video quality. C8 represents the constraint of bit error rate, which guarantees the transmission quality.

3. Resource Allocation Criteria

In this section, we introduce the way to get the accurate mathematical models of optimization criterion F_m^{task} , which describe the mathematical relationship between the performance of intelligent tasks and communication parameters. Besides, these mathematical models are the basis for resource allocation problems. The steps of optimization criterion modeling are described as follows.

First is related data generation. In order to get the final mathematical model as $y=f(x)$, we first need to obtain a large number of data as (x, y) , where x represents communication parameters and y represents the performance values of task completion. Due to communication progress involving two processes, encoding and transmission, in which compression and noise are the key factors leading to data quality decline and affecting the performance of subsequent tasks, in this paper, x represents different compression rates q_m and bit error rates p_m . Based on the proposed communication system, multimedia data is transmitted at different compression rates and bit error rates, and then, intelligent

tasks are completed at the edge server to obtain the performance values such as accuracy and mAP of corresponding tasks.

Second is mathematical model selection. Based on the data obtained in the first step, scatter diagrams are drawn to analyze the trend, as shown in Figures 3 and 4. Then, according to the experimental results, the power function model is selected in this paper. Firstly, the power function model can reflect the corresponding relationship well, and secondly, it has monotonicity and is convenient for derivation.

Third is model parameter solution. Based on the large amount of data obtained in the first step and the power function model selected in the second step, mean square error (MSE) criterion is taken as the guidance to design the algorithm for solving model parameters. The flow is shown as Algorithm 1.

According to the proposed method, we obtain the models for different AI tasks, such as classification, detection, and Re-ID. The models reveal the relationship between task performance and communication parameters like compression rate or bit error rate. The mathematical formulas for the criteria models are as follows:

$$F_m^{\text{task}}(q_m) = a_1 \cdot q_m^{a_2} + a_3, \quad (7)$$

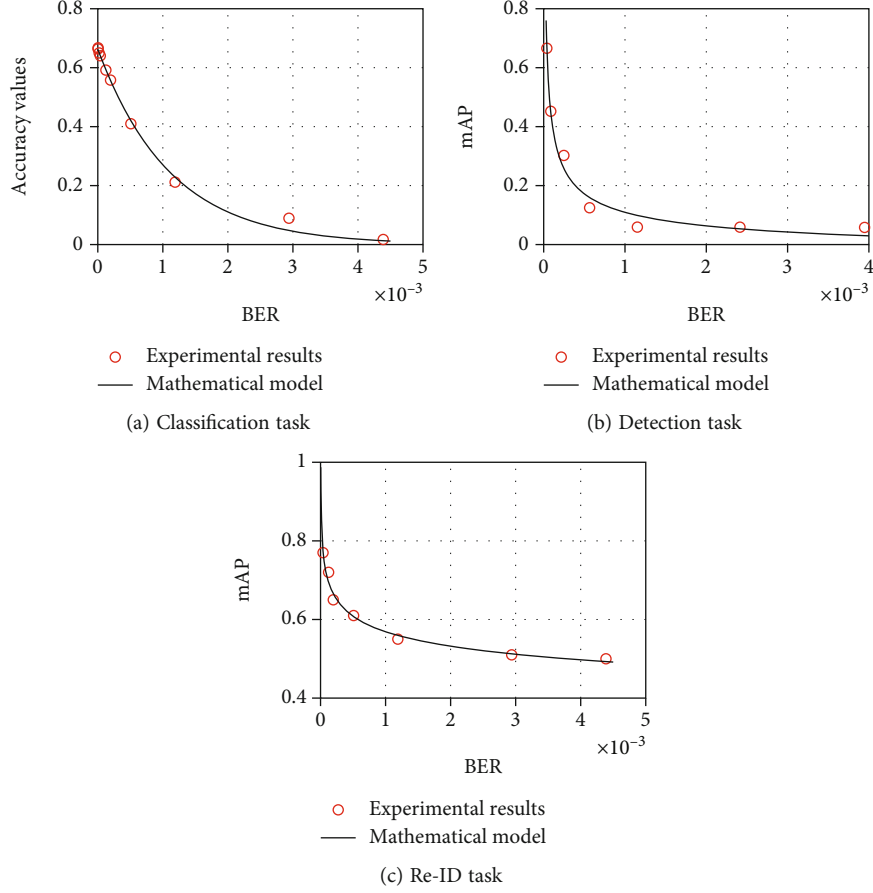


FIGURE 4: The relationship between task performance and bit error rate with different tasks.

```

1: Input: Initial parameters  $\mathbf{a} = [a_1, a_2, a_3]$ ,  $\mathbf{b} = [b_1, b_2, b_3]$ , data set  $D$ , step length  $\delta$ , and threshold  $L_0$ ;
2: Output: Parameters  $\mathbf{a}$ ,  $\mathbf{b}$ .
3: while  $L_1(\mathbf{a}) \geq L_0$  or  $L_2(\mathbf{b}) \geq L_0$  do
4:    $(q_m, F_m^{\text{task}}(q_m)^*), (p_m, F_m^{\text{task}}(p_m)^*) \in D$  do
5:      $F_m^{\text{task}}(q_m) = a_1 \cdot q_m^{a_2} + a_3$ 
6:      $F_m^{\text{task}}(p_m) = b_1 \cdot p_m^{b_2} + b_3$ 
7:   end for
8:   Compute the loss based on MSE:
9:      $L_1(\mathbf{a}) = 1/2D \sum_{d=1}^D (F_m^{\text{task}}(q_m) - F_m^{\text{task}}(q_m)^*)^2$ 
10:     $L_2(\mathbf{b}) = 1/2D \sum_{d=1}^D (F_m^{\text{task}}(p_m) - F_m^{\text{task}}(p_m)^*)^2$ 
11:   Compute the gradient of  $\mathbf{a}$  and  $\mathbf{b}$ :
12:    $G_1(\mathbf{a}) = \partial L_1(\mathbf{a}) / \partial \mathbf{a}$ ,  $G_2(\mathbf{b}) = \partial L_2(\mathbf{b}) / \partial \mathbf{b}$ 
13:   Gradient descent based update strategy:
14:    $\mathbf{a} := \mathbf{a} - \delta G_1(\mathbf{a})$ ,  $\mathbf{b} := \mathbf{b} - \delta G_2(\mathbf{b})$ 
15: end while

```

ALGORITHM 1: Gradient Descent algorithm of resource allocation criterion modeling.

$$F_m^{\text{task}}(p_m) = \frac{b_1}{p_m^{b_2}} + b_3, \quad (8)$$

where F_m^{task} denotes the task performance like accuracy and mAP, q_m and p_m are the compression rate and bit error rate,

respectively, and the other symbols are model parameters, which are solved by Algorithm 1.

In the flowchart, \mathbf{a} and \mathbf{b} represent the vectors of model parameters. The loss functions L_1 and L_2 represent the MSE values of $F_m^{\text{task}}(q_m)$ and $F_m^{\text{task}}(p_m)$, respectively, and L_0 is the MSE threshold. Moreover, G_1 and G_2 represent the

derivatives of loss functions L_1 and L_2 , respectively, which are used to iteratively update the solutions of model parameters.

4. Resource Allocation Algorithm

According to the above analysis, we have gotten the models to describe the relationship between the AI task performances and communication metrics. To get the AI tasks done better in the edge server, reliable resource allocation should be carried out in the process of transmission, which means to maximize the accuracy or mAP performance. Hence, in this section, we first analyze the difficulties in solving the above optimization problem. Then, we propose a multiagent deep Q-network (MADQN-) based algorithm to solve the problem.

4.1. Optimization Problem Analysis. Based on the above optimization problem, the proposed system transmission model and the scenario, the design of algorithm faces the following challenges.

4.1.1. Nonconvexity Problem. For the optimization problem, its objective function $F_m^{\text{task}}(q_m, p_m)$ is nonlinear, and its constraint conditions C4 and C8 are nonconvex and nonlinear (which can be derived by Equation (4), and its optimization variables are multidimensional. Therefore, the optimization problem P1 is essentially a NP-hard problem, which is difficult to be solved by convex optimization methods.

4.1.2. Dynamic Transmission Conditions. For the proposed transmission model, the channel gain h_m is dynamically changing, which leads to the traditional convex optimization algorithms, and heuristic algorithms are difficult to capture the dynamic characteristics of the channel.

4.1.3. High Requirements in the Scenario. For the proposed task-oriented dynamic vehicular network scenario, the complexity and stability of the algorithm are highly required.

Recently, deep reinforcement learning has shown strong advantages in solving resource allocation optimization problems in dynamic environments. Therefore, this paper considers using deep reinforcement learning to solve the challenges mentioned above, mainly based on the following.

By transforming the original nonconvex mathematical problems into sequential decision problems, deep reinforcement learning uses historical data and interaction with the environment to learn strategies and uses neural networks to approximate the optimal solution, so as to solve the NP-hard problems. One of the advantages of deep reinforcement learning is to solve the problems of dynamic environment [26], which refers to randomness under fixed state distribution, such as the channel model h_m proposed in this paper. The neural network fitting function itself can map the close set to the close domain so as to improve the robustness of dynamic learning. Another advantage of deep reinforcement learning lies in its strong generalization ability [27]. Through full offline training, online decision-making can greatly reduce the complexity, which fits well with the proposed transmission system proposed, as shown in Figure 2. In

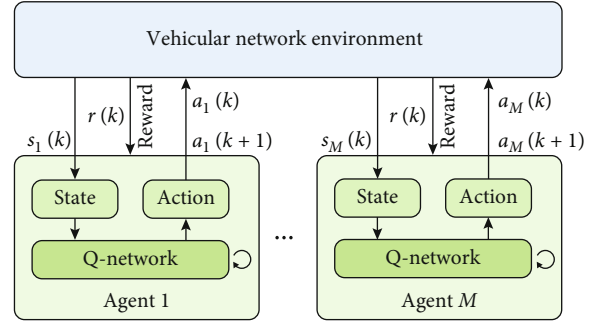


FIGURE 5: The proposed DQN algorithm flowchart.

addition, introducing experience replay and iterative updating mechanism can also improve the convergence stability of deep reinforcement learning.

4.2. Key Elements for Deep Reinforcement Learning. The DQN is a widely used deep reinforcement learning algorithm. Figure 5 is the flowchart of the proposed DQN algorithm in this paper, in which all vehicles act as agents to optimize the objective function in the optimization problem P1, namely, the performance metrics of the intelligent tasks. After performing an action $a(k)$, the agents can receive feedback information such as reward values $r(k)$ related to tasks' performance from the environment, update the current state $s(k)$, and then select the next action $a(k+1)$ according to the current state and action selection strategy by Q-network, until the convergence of the best strategy is obtained.

There are some key elements in the design process of DQN algorithm, especially to the optimization problem and system model in this paper.

4.2.1. Agent. We regard M vehicles in a vehicular network as the agents in the multiagent DQN algorithm.

4.2.2. Action. In the proposed scheme, the execution action contains three dimensions, namely, the change of the encoding packet length L_m , the bandwidth resource B_m , and the transmission power P_m of the m -th vehicle. There are three options for actions (increases, remains, or decreases) in each dimension. For example, $L_A = \{+\Delta L, 0, -\Delta L\}$ represents the change of the allocated packet length; the same goes for the changes of allocated bandwidth B_A and power P_A . Action space is represented as

$$A_m = \{\mathbf{a}_m | \mathbf{a}_m \in \{L_A \times B_A \times P_A\}\}, \quad (9)$$

where \times denotes Cartesian product. Therefore, the action space size is $3 \times 3 \times 3$, and its dimension is 3. For example, $\mathbf{a}_m = (+, -, 0)$ indicates that the current action is as follows: the packet length allocated to the m -th vehicle increases, the bandwidth decreases, and the power remains unchanged. Where considering that the packet length $L_m \in N^*$, the change of L_m is set as $\Delta L = 1$. In the discretization of B_m and P_m , $\Delta B = 0.001B_{\max}$ and $\Delta P = 0.001P_{\max}$, respectively.

4.2.3. State. We set the state of observation as the allocated resources of vehicles, as well as the channel state information

obtained from the environment, so the state space is expressed as

$$S_m = \{s_m | s_m \in \{L_m, B_m, P_m, h_m\}\}, \quad (10)$$

where the dimension of the state space S_m is 4.

4.2.4. Reward. The objective function is that all users execute the action strategy to maximize the total task performance under constraint conditions. To achieve this purpose, in the case of packet length, bandwidth and power changes, and channel state changes, the reward function is set as

$$r = \begin{cases} \sum_{m=1}^M F_m^{\text{task}}, & \text{if C1} \sim \text{C8 are met,} \\ -10, & \text{otherwise.} \end{cases} \quad (11)$$

4.2.5. Q-Network. Figure 6 shows the Q-network structure. Each agent updates the network structure by using the data acquired by itself. It is worth noting that, considering the possible lack of computing power of the agent, the agent can upload the data to the server, and after the server completes the training, the network parameters are downlink transmitted to the agent. As shown in Figure 6, the Q-network contains three layers, namely, the input layer, the hidden layer, and the output layer.

The input dimensions are equal to the sum of the of state vector s_m dimensions, the hidden layer dimensions are H , and the output layer dimensions are equal to the action vector a_m dimensions, respectively. In addition, we use rectified linear unit (ReLU) as the activation function. For simplicity, the Q-function of agent m is expressed as Q_m . To train the DQN, we use a finite-size experience replay buffer Z to save the history transition samples $z(k) = (s(k), a(k), r(k), s(k+1))$, and old samples will be discarded when the storage is full. The two DQNs evaluation Q-network $Q_\theta(\cdot)$ and target Q-network $Q_{\theta^*}(\cdot)$ are used to approximate Q-function, with θ and θ^* being their weights, have the same structure. Here, θ is expressed as

$$\theta = (\omega_{\text{in}}, \mathbf{b}_{\text{in}}, \omega_{\text{out}}, \mathbf{b}_{\text{out}}), \quad (12)$$

where ω_{in} and \mathbf{b}_{in} are parameters of the first fully connected layer from input layer to hidden layer and ω_{out} and \mathbf{b}_{out} are parameters of the second fully connected layer from hidden layer to output layer, as shown in Figure 6. Moreover, the weight update of $Q_{\theta^*}(\cdot)$ is expressed as slowly approaching the weight of $Q_\theta(\cdot)$, as $\theta^* \leftarrow \theta$, after C steps.

4.3. Multiagent DQN Algorithm. Algorithm 2 shows the process of multiagent DQN Algorithm. The algorithm mainly includes two parts: initialization process and reinforcement learning process.

Reinforcement learning is a process of repeated iteration; each iteration should solve two problems: give a strategy to obtain value function, and update the strategy according to

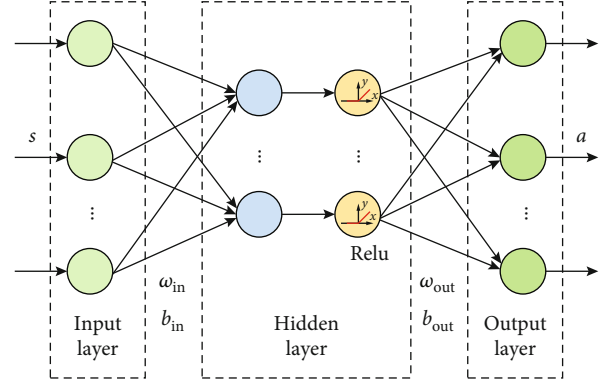


FIGURE 6: The proposed Q-network structure.

the value function. The iterative process is as follows: (1) The environment gives an observed state s , and the agent obtains all $Q_\theta(s, a)$ of state s based on the value function network, namely, Q-network $Q_\theta(\cdot)$. Then, the agent selects actions and makes decisions using ϵ -greedy strategy. (2) Upon receiving the action, the environment will give a reward and the next observation state. (3) The agent update the parameters θ of Q-network according to the loss function, and then enter the next step. (4) The cycle continues until a convergent Q-network is trained.

In the proposed DQN algorithm, experience replay mechanism and target network are introduced: (1) Experience replay is used to solve the problem of data correlation, that is, to store the experienced data in a buffer and extract a part of the data from the buffer for each parameter updating, so as to ensure that the training samples are independent and equally distributed, improve the utilization rate of data, and make the model better converge. (2) Target network is used to solve the problem of value function fluctuation in the iterative process. By using target network, the model calculating the target value will be fixed in a period of time, which can reduce the volatility of the model and make the training more stable.

In addition, the training effect of DQN algorithm is related to its main parameters. After many trials, the main parameters of the algorithm in this paper are set as follows: the learning factor μ is set at 0.001, so that the algorithm retains most of the historical training results and pays more attention to past experience, and the discount factor λ is set as 0.9, allowing the algorithm to consider 90% of the next reward and pay more attention to long-term rewards. The coefficient of ϵ -greedy strategy is set as $\epsilon = 1/\sqrt{k}$, so that the algorithm strikes a balance between exploration and exploitation.

4.4. Convergence and Complexity. First, we discuss the convergence of the proposed algorithm: The convergence of DQN is difficult to prove directly theoretically, but the training of DQN is stable due to the introduction of the mechanism of experience replay and target network. It can be proved that network parameters can converge to a very small interval, so that a stable approximate optimal solution can be obtained through DQN. The proof process is shown in Appendix A.


```

1: Input: action space  $A$ , state space  $S$ , learning factor  $\mu$ , discount factor  $\lambda$ , and other system parameters;
2: Output: Target Q-network  $Q_{\theta^*}(\cdot)$ .
3: Initialization Process:
4: for each agent  $m$  do
5:   Initialize replay buffer  $Z_m$ ;
6:   Initialize the weights of Q-network  $\theta$ ;
7:   Initialize the weights of target Q-network  $\theta^* = \theta$ .
8: end for
9: Learning Process:
10: for each agent  $m$  do
11:   repeat
12:     Initialization state  $\mathbf{s}$ 
13:     for  $k = 1, K$  do
14:       Generate a random number  $x_k \in [0, 1]$ ;
15:       if  $x_k < \varepsilon$  then
16:         Select a random action  $\mathbf{a}$  from  $A$ ;
17:       else
18:         Select the action  $\mathbf{a} = \arg \max Q_{\theta}(\mathbf{s}, \mathbf{a})$ .
19:       end if
20:       Perform action  $\mathbf{a}$ ;
21:       Get new state  $\mathbf{s}'$  and reward  $r(k)$  by Equation (11);
22:       Store transition  $\mathbf{z}(k) = (\mathbf{s}, \mathbf{a}, r, \mathbf{s}')$  in  $Z$ ;
23:       Sample random transitions  $(\tilde{\mathbf{s}}, \tilde{\mathbf{a}}, \tilde{r}, \tilde{\mathbf{s}}')$  from  $Z$ ;
24:       
$$y = \begin{cases} \tilde{r}, & \text{if } \tilde{\mathbf{s}}' \text{ is the termination state;} \\ \tilde{r} + \gamma \max Q_{\theta^*}(\tilde{\mathbf{s}}', \tilde{\mathbf{a}}'), & \text{otherwise.} \end{cases}$$

25:       Use  $L(\theta) = (y - Q_{\theta}(\tilde{\mathbf{s}}, \tilde{\mathbf{a}}))^2$  as a loss function to train Q-network;
26:       Update the weights:  $\theta \leftarrow \theta - \mu \cdot \nabla \theta$ ;
27:        $\mathbf{s} \leftarrow \mathbf{s}'$ ;
28:        $\theta^* \leftarrow \theta$ , after  $C$  steps.
29:     end for
30:   until  $Q_{\theta^*}(\cdot)$  converges.
31: end for

```

ALGORITHM 2: Multiagent DQN algorithm.

Second, we discuss the complexity of the proposed algorithm: The complexity of DQN can be divided into two parts: training complexity and inference complexity. The training complexity is related to the number of back propagation and iterations. Each iteration requires one back propagation, and one back propagation requires four parallel derivative operations. Assuming that the time complexity of a derivative operation is $O_{\partial}(1)$, the time complexity of the training stage is

$$O_{\text{Training}} = O_{\partial}(4 * K). \quad (13)$$

Inference complexity is mainly related to Q-network structure. As shown in Figure 6, each inference needs to go through two fully connected layers, so the computational complexity in the reasoning stage can be expressed by the required multiplication times, which is

$$\begin{aligned} O_{\text{Inference}} &= (d_{\text{in}} * d_{\text{h}} + d_{\text{h}}) + (d_{\text{h}} * d_{\text{out}} + d_{\text{out}}) \\ &= (d_{\text{in}} + d_{\text{out}} + 1)d_{\text{h}} + d_{\text{out}}, \end{aligned} \quad (14)$$

where $d_{\text{in}}, d_{\text{h}}, d_{\text{out}}$ are dimensions of input layer, hidden

layer, and output layer, respectively, and the dimensions represent the number of neurons of each layer. The consumed time of online resource allocation depends on the inference complexity of the proposed DQN algorithm. The proposed multiagent DQN is distributed, its training complexity is only related to the number of iterations, and its inference complexity is only related to the dimensions of each layer, which are fixed. The complexity will not expand exponentially with the increase of vehicles. Therefore, the proposed algorithm is suitable for scenarios with low delay and high access.

5. Numerical Results

In this section, numerical results are provided for the performance evaluation of the proposed algorithms. Firstly, the parameter settings of communication system simulation and intelligent task experiment are introduced. Secondly, the datasets used in this paper and the related schemes are introduced. Thirdly, the performance evaluation results of Algorithm 1 are presented and analyzed. Lastly, the performance evaluation results of Algorithm 2 are presented and analyzed.

5.1. Parameter Setting. The communication simulation system is constructed based on Simulation of Urban Mobility (SUMO), Matlab R2019a, and Pycharm 2019.1.1 platform. Firstly, the simulation scenarios of vehicular networks are based on the urban intersection scene generated by SUMO. Each road includes 4 two-way lanes 3.5 m wide, and the initial positions of vehicles are randomly generated. Secondly, the Winner model in the 3GPP TR 36.885 standard is used for the channel model, as described in Section 4. Meanwhile, the small scale fading caused by vehicle movement and construction is considered, and the time-decrement scale fading model shown in Equation (5) is adopted. The specific system simulation parameter settings are based on 3GPP TR 36.885, as shown in Table 1. Lastly, considering the time-varying unsteady characteristics of the vehicular networks in practice, 200 independent Monte Carlo simulations are used in each experiment to take the average value to eliminate errors caused by abnormal data

In the experiments of intelligent tasks, the GPU model is NVIDIA GeForce RTX 3090, and the training and testing environment is Windows 10 + CUDA 10.2. The intelligent tasks adopted in this paper include classification, object detection, and Re-ID. Deep learning based experimental parameter settings are shown in Table 2, mostly based on experience. In addition, JPEG 2000 is used for image compression and coding, and HEVC is used for video compression and coding.

5.2. Datasets and Related Scheme Introduction. The datasets used in this paper include four existing datasets and the new dataset constructed in this paper. The datasets are summarized in Table 3. The STL-10 [28] dataset mainly includes image data and is applicable to classification tasks. Caltech [29] and Waymo [30] datasets are capture by real vehicular cameras in driving scenarios, which are mainly used for object detection tasks in vehicular networks. Market-1501 [31] mainly contains image data and is suitable for Re-ID tasks. In addition, a semantic communication-oriented dataset [32], namely, SCO, containing 5100 images and 10 video clips, is also constructed for classification tasks and object detection tasks.

To verify the performance advantages of the proposed resource allocation scheme, three comparison schemes are used. Based on the main characteristics of each scheme, the scheme names are all abbreviated. The schemes are detailed as follows:

- (i) TPO-JRA scheme (Task Performance-oriented Joint Resource Allocation scheme): it is the proposed scheme in this paper, which is aimed at maximizing the performance of intelligent task, and the optimization variables include data rate resource, bandwidth resource, and power resource. The optimization algorithm is based on MA-DQN
- (ii) CPO-RA scheme (Content Priority Oriented Resource Allocation Scheme): in this scheme, the optimization objective is to maximize the effective information, the definition of which is shown in

TABLE 1: Communication system parameters.

Parameters	Values
The edge server coverage	0.5 km
Number of vehicles M	3 ~ 18
Range of vehicle speed $v_m, \forall m$	0 ~ 60 km/h
The modulation order N	4
Noise power spectral density n_0	-174 dBm/Hz
Band mid-frequency f_c	2GHz
The time interval of small-scale fading t_e	50 ms
The packet size G_m	20 KB
The video block size S_m	400 KB
The video block duration f_c	0.1 s
The total data rate R_{\max}^B	200 ~ 700 Kbps
The total bandwidth B_{\max}	0.1 ~ 0.9 MHz
The total power P_{\max}	5 ~ 45 dBm
The minimum performance limit f_{\min}	0.1
The delay threshold range τ	50 ~ 300 ms
The compression rate threshold q_e	0.95
The bit error rate threshold p_e	0.001

[20]. The optimization variables include power resource and bandwidth resource, and the optimization algorithm is based on Q-learning

- (iii) QoC-RA scheme (Quality of Content-based Resource Allocation scheme): in this scheme, the optimization objective is to maximize content quality, which is defined in [15]. Under the object detection task, that is, the average detection accuracy, the optimization variable is bit-rate resource, and the adopted optimization algorithm is based on convex optimization
- (iv) MRA scheme (Mean Resource Allocation scheme): in this scheme, all resources are evenly allocated to each user. This scheme serves as a baseline for other schemes to verify the performance gain of the proposed scheme

In addition, in order to ensure the fairness of comparison, the same algorithm is adopted in this paper under the same scenario and intelligent task.

5.3. Performance Results of Model Solving Method. Figures 3 and 4 show the relationship between task performance and compression rate and bit error rate, respectively. Task performance decreases with the increase of compression rate and bit error rate. This is because the semantic or content of the original data is lost due to lossy compression or noise interference, so that the machine at the edge server cannot correctly identify the semantic or content of the original data through deep learning networks. Secondly, the relationship between task performance and compression rate presents the nature of concave function, while the relationship

TABLE 2: Deep learning-based experiment parameters.

Parameters	Values
The learning factor μ	0.001
The discount factor γ	0.9
The update frequency of target Q-network C	5
The maximum number of iterative steps K	10^4
The size of replay buffer Z_m	200
The dimensions of input layer d_{in}	4
The dimensions of hidden layer d_h	10
The dimensions of output layer d_{out}	3

TABLE 3: The adopted datasets for intelligent tasks.

Tasks	Networks	Datasets
Classification	ResNet-18	STL-10,SCO
Re-ID	ResNet-50	Market-1501
Detection	Faster-RCNN	Caltech, Waymo, SCO

between task performance and bit error rate presents the nature of convex function. This is because they have different ways of semantic distortion. The lossy compression mainly leads to blur distortion, and the noise mainly causes the symbol errors in the decoding process. In addition, the curves of each task have differences, which is because different tasks are based on diverse deep learning networks, which have various degrees of sensitivity to compression distortion and noise.

Table 4 shows the numerical results of model parameters and RMSE metric, and all numerical results are reserved for 4 significant digits. RMSE reflects the degree of agreement between the predicted values based on the mathematical model and the actual values. For the proposed method, in the classification, detection, and Re-ID tasks, the RMSE values of model Equation (7) are 0.03438, 0.04303, and 0.02984, respectively, and the RMSE values of model Equation (8) are 0.04078, 0.03739, and 0.01633, respectively. All of them are less than 0.05, reflecting that the modeling method (Algorithm 1) has good accuracy performance and can reflect the relationship between task performance and system variables.

5.4. Performance Results of Resource Allocation Scheme. In this section, we use the task performance values related to F_m^{task} in Equation(1) to verify the advantages of the proposed scheme. The experiments include three different intelligent tasks, classification, detection, and Re-ID tasks. A different task is related to different metric, such as accuracy or mAP, but the values are between 0 and 1. Hence, we use the weighted average values of different vehicles with different kinds of tasks to present the performance metric of the whole system. The larger the values are, the better intelligent tasks of the system are completed.

5.4.1. Performance versus Resource Parameters. Figures 7–9 show the curves of task performance versus bit rate, band-

TABLE 4: The results of model solution.

Parameters	Classification	Detection	Re-ID
a_1	-1.017	-0.7849	-1.155
a_2	3.551	2.699	4.908
a_3	0.7387	0.7940	0.7606
b_1	-6.420	0.008075	0.3320
b_2	-0.3953	0.4442	0.08931
b_3	0.7364	-0.06427	-0.04626
RMSE- a	0.03438	0.04303	0.02984
RMSE- b	0.04078	0.03739	0.01633

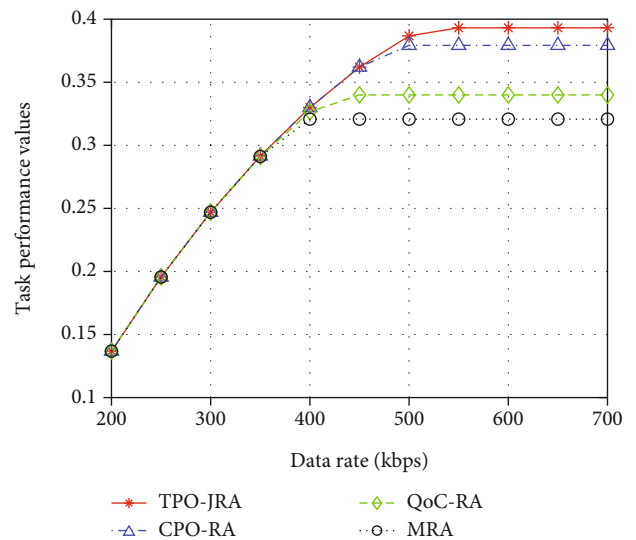


FIGURE 7: Task performance versus data rate resource.

width, and power, respectively. With the variation of resource parameters, the proposed resource allocation scheme has the best performance, which verifies the effectiveness of the proposed scheme. Combined with Figures 7–9, the following conclusions can be drawn: Firstly, optimization of resource allocation is crucial to the performance improvement of intelligent tasks, which indicates that in intelligent task-oriented communication system, resource optimization can further ensure the correct understanding of transmitted data by the server, rather than only pursuing the improvement of computing capacity of the server. Secondly, when resources are limited, it is necessary to jointly optimize different types of resource parameters, because the influence of various resource parameters on video quality is coupled in the transmission process.

In addition, within the numerical range of experimental settings, the performance gains brought by optimization of different resource parameters are inconsistent, which is due to the different mathematical relationship between resource parameters and the objective function. There is an inequality of average performance gains with resource parameters, $\Delta\bar{F}(P_m) > \Delta\bar{F}(L_m) > \Delta\bar{F}(B_m)$, which indicates that in the actual environment, the allocation of power should be given

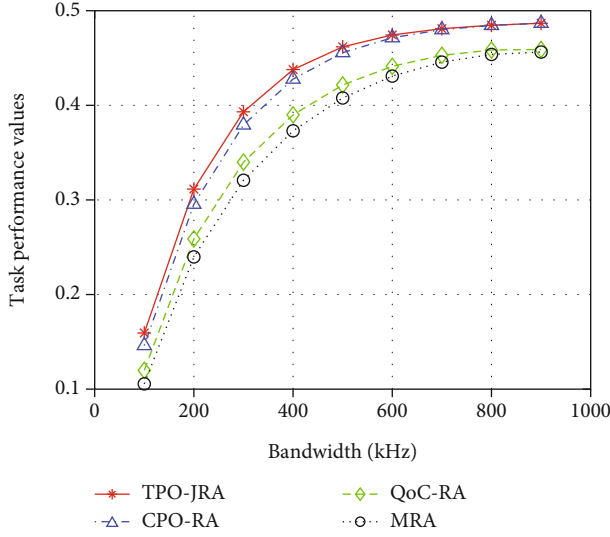


FIGURE 8: Task performance versus bandwidth resource.

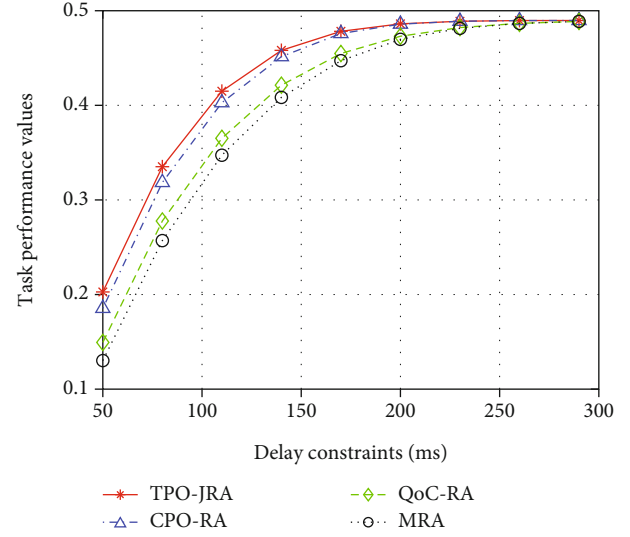


FIGURE 10: Task performance under different delay constraints.

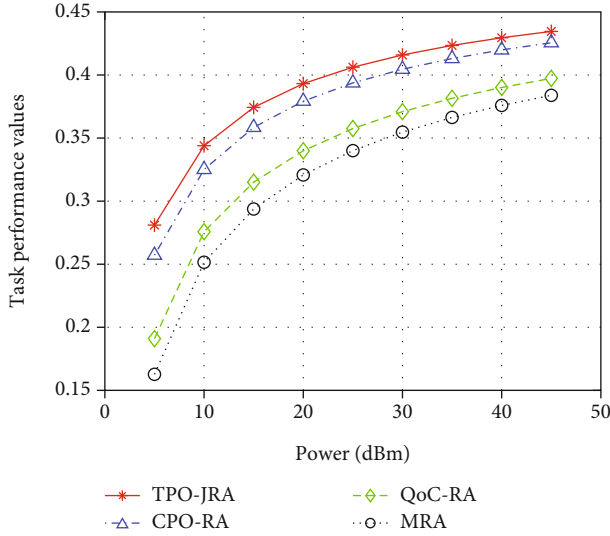


FIGURE 9: Task performance versus power resource.

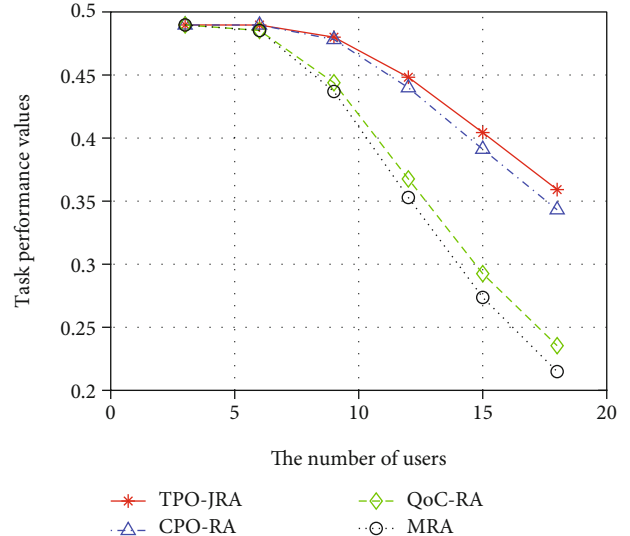


FIGURE 11: Task performance under different number of vehicles.

priority. Lastly, with the increase of resources, the performance gain decreases gradually, because under the condition of resource saturation, the performance tends to be stable, which is determined by the neural network structure related to intelligent tasks.

5.4.2. Performance versus Delay and Number of Vehicles. With different schemes, the change of task performance under the time-delay constraint is shown in Figure 10. Under the condition of strict delay constraints, the performance of the proposed scheme has advantages, which is due to the reasonable design of the constraints in the optimization problem. The proposed algorithm explores the best action strategy to maximize the task performance. At the same time, the delay requirements are met, according to Equation (11). Secondly, the complexity of the proposed algorithm is low. Based on the above analysis, the proposed

resource allocation scheme is more suitable for time-critical scenarios.

The change of task performance with the number of vehicles is shown in Figure 11. It can be seen that with the increase of the number of vehicles connected to the edge server, the overall performance shows a downward trend, because the increase of vehicles causes more intense competition for resources, and the average resources that can be allocated to each vehicle decreases. However, with the increase of the number of vehicles, the proposed resource allocation scheme still maintains the optimal performance, which verifies the performance advantage of the proposed scheme in the scenario of large access or resource shortage. In addition, it also reveals that large-scale access systems must optimize the multidimensional parameters to ensure the accurate understanding of the data at the receiving end.

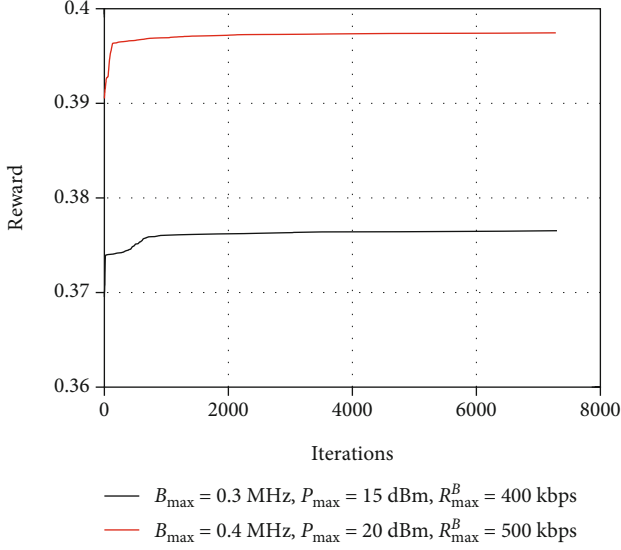


FIGURE 12: Convergence curve of the proposed algorithm.

5.4.3. Convergence Performance. The convergence results of the proposed resource allocation algorithm are shown in Figure 12. To get rid of burrs, the convergence curves are smoothed for better visual effect. More precisely, the value of each step is the average of the near 5 original steps. The smoothing operation does not affect the convergence performance of the algorithm. It can be seen that the proposed algorithm converges stably within a limited number of iterations. There are two reasons for this. First, the introduction of experience replay mechanism and target network enables the parameters of Q -network to converge. Second, the design of action selection strategy makes the algorithm pay more attention to learning historical experience rather than exploring when the number of iterations increases, which ensures the stability of Q -network. Moreover, according to the curves, our algorithm seems to converge very fast, that is, because the structure of Q -network we designed is relatively simple as shown in Figure 6, which only contains three layers, and the dimensions of each layer are 4, 10, and 3, respectively. This results in fast convergence of the algorithm. Because, in each update iteration, the computation of network parameters becomes smaller, as shown in Equation (13) and (14). The choice of network structure and layer dimensions are the results of experience.

6. Conclusion

This paper studied resource allocation schemes, in order to achieve efficient transmission of multimedia data and accurate completion of intelligent tasks in the IoV scenarios. In this paper, a mathematical model of resource allocation optimization criterion for intelligent tasks was constructed, and the model parameters were solved by a gradient descent algorithm. For classification, detection, and Re-ID tasks, the RMSE index of the algorithm was less than 0.05. Secondly, under the guidance of this model, this paper designed a joint allocation algorithm of data rate, bandwidth, and power resources based on MADQN and discussed the con-

vergence and complexity of the algorithm. Experimental results showed that the proposed resource allocation scheme is more suitable for the intelligent networked environment with lots of computer vision tasks. The results revealed that to improve the performance of intelligent tasks; not only the intelligent algorithms but also the communication technologies like resource allocations should be considered.

In line with the general trend of the close combination of communication technology and artificial intelligence, the scheme proposed in this paper provides a new solution to the difficult problems existing in the era of intelligent driving, such as complex traffic environment and low transmission efficiency. In the future work, we will deduce a more universal resource allocation criterion from the perspective of information theory and analyze the performance of intelligent tasks oriented communication system from the theoretical perspective.

Appendix

A. Proof of the Convergence of Q -Network

In the proposed MADQN algorithm, parameters update and follow the rule:

$$\theta = \theta - \alpha \cdot \nabla \theta, \quad (\text{A.1})$$

where α is the learning factor and $0 < \alpha < 1$ and $\nabla \theta$ denotes the partial derivative of the value function with respect to θ , which is expressed as

$$\begin{aligned} \nabla \theta &= \frac{\partial L(\theta)}{\partial \theta} = 2 \left(Q(a') - Q_\theta(a) \right) \left(-\frac{\partial Q_\theta(a)}{\partial \theta} \right) \\ &= -2 \left(Q(a') - Q_\theta(a) \right) a. \end{aligned} \quad (\text{A.2})$$

Hence, by replacing the true value function with the current estimated value function, we can get the parameter update rule, which is expressed as

$$\theta = \theta + \alpha \left(r + \gamma \theta a' - \theta a \right) a. \quad (\text{A.3})$$

After k iterations, the parameters can be expressed as

$$\theta_k = \theta_k + \alpha \left\{ (1 - \alpha)^k \theta_k a + \left[1 - (1 - \alpha)^k \right] \left(r + \gamma \theta_k a' \right) - \theta_k a \right\} a. \quad (\text{A.4})$$

Due to $0 < (1 - \alpha) < 1$, when k is big enough, there is $(1 - \alpha)^k \rightarrow 0$; thus,

$$\theta_k = \theta_k + \alpha \left(r + \gamma \theta_k a' - \theta_k a \right) a = \theta_k - \alpha \cdot \nabla \theta_k. \quad (\text{A.5})$$

That is, after k iterations, the parameters' updating form is still the original form of Behrman equation. Based on the conclusion of [33], the parameters' updating of the proposed algorithm tends to converge.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

A preliminary work of this paper has been presented as Arxiv in Cornell University according to the following link: <https://arxiv.org/abs/2009.13379>.

Conflicts of Interest

The authors declare that there is no conflict of interest.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities under Grant No. 2021XD-A01-1, the Beijing Natural Science Foundation under Grant No. 4202049, and the Industrial Internet Research Institute (Jinan) of Beijing University of Posts and Telecommunications under Grant No. 201915001.

References

- [1] R. Hussain and S. Zeadally, "Autonomous cars: research results, issues, and future challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1275–1313, 2019.
- [2] J. Zhang, F. Wang, K. Wang, W. H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.
- [3] B. Ranft and C. Stiller, "The role of machine vision for intelligent vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 8–19, 2016.
- [4] J. Zhang and K. B. Letaief, "Mobile edge intelligence and computing for the Internet of vehicles," *Proceedings of the IEEE*, vol. 108, no. 2, pp. 246–261, 2020.
- [5] J. W. Kim and D. K. Jeon, "A cooperative communication protocol for QoS provisioning in IEEE 802.11p/wave vehicular networks," *Sensors*, vol. 18, no. 11, p. 3622, 2018.
- [6] Z. Wu, Z. Lu, P. C. Hung, S. C. Huang, Y. Tong, and Z. Wang, "QaMeC: a QoS-driven IoVs application optimizing deployment scheme in multimedia edge clouds," *Future Generation Computer Systems*, vol. 92, pp. 17–28, 2019.
- [7] L. Sun, H. Shan, A. Huang, L. Cai, and H. He, "Channel allocation for adaptive video streaming in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 734–747, 2016.
- [8] H. Zhang, Y. Ma, D. Yuan, and H. H. Chen, "Quality-of-service driven power and sub-carrier allocation policy for vehicular communication networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 197–206, 2011.
- [9] L. Liang, G. Y. Li, and W. Xu, "Meeting different QoS requirements of vehicular networks: a D2D-based approach," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3734–3738, New Orleans, LA, USA, 2017.
- [10] X. Tao, Y. Duan, M. Xu, Z. Meng, and J. Lu, "Learning QoE of mobile video transmission with deep neural network: a data-driven approach," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1337–1348, 2019.
- [11] A. K. Bairagi, S. F. Abedin, N. H. Tran, D. Niyato, and C. S. Hong, "QoE-Enabled unlicensed Spectrum sharing in 5G: a game theoretic approach," *IEEE Access*, vol. 6, pp. 50538–50554, 2018.
- [12] M. Jalil Piran, N. H. Tran, D. Y. Suh, J. B. Song, C. S. Hong, and Z. Han, "QoE-driven channel allocation and handoff management for seamless multimedia in cognitive 5G cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6569–6585, 2017.
- [13] H. Zhu, Y. Cao, W. Wang, B. Liu, and T. Jiang, "QoE-aware resource allocation for adaptive device-to-device video streaming," *IEEE Network*, vol. 29, no. 6, pp. 6–12, 2015.
- [14] H. Gu, Y. Dong, and T. Cao, "Data driven QoE-QoS association modeling of conversational video," *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, vol. 2019, pp. 1–4, 2019.
- [15] X. Chen, J. Hwang, D. Meng, K. H. Lee, R. L. de Queiroz, and F. M. Yeh, "A quality-of-content-based joint source and channel coding for human detections in a mobile surveillance cloud," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 1, pp. 19–31, 2017.
- [16] E. C. Strinati and S. Barbarossa, "B6G networks: beyond Shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, article 107930, 2021.
- [17] G. Shi, Y. Xiao, Y. Li, and X. Xie, "From semantic communication to semantic-aware networking: model, architecture, and open problems," *IEEE Communications Magazine*, vol. 59, no. 8, pp. 44–50, 2021.
- [18] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.
- [19] H. Xie and Z. Qin, "A lite distributed semantic communication system for internet of things," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 142–153, 2021.
- [20] J. Chen, C. Guo, C. Feng, M. Zhu, and Q. Sun, "Content driven and reinforcement learning based resource allocation scheme in vehicular network," in *IEEE International Conference on Communications (ICC)*, pp. 1–6, Montreal, QC, Canada, 2021.
- [21] M. Zhu, C. Feng, J. Chen, C. Guo, and X. Gao, "Video semantics based resource allocation algorithm for spectrum multiplexing scenarios in vehicular networks," in *IEEE/CIC International Conference on Communications in China (ICCC Workshops)*, pp. 31–36, Xiamen, China, 2021.
- [22] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.
- [23] B. Sklar, *Digital Communication Fundamentals and Applications*, Prentice Hall, Upper Saddle River, New Jersey, USA., 2001.
- [24] C. Guo, L. Liang, and G. Y. Li, "Resource allocation for vehicular communications with low latency and high reliability," *IEEE Transactions on Wireless Communications*, vol. 18, no. 8, pp. 3887–3902, 2019.
- [25] J. Shi, Z. Yang, H. Xu, M. Chen, and B. Champagne, "Dynamic resource allocation for LTE-based vehicle-to-infrastructure

- networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 5017–5030, 2019.
- [26] N. C. Luong, D. T. Hoang, S. Gong et al., “Applications of deep reinforcement learning in communications and networking: a survey,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019.
- [27] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, “Deep reinforcement learning for multiagent systems: a review of challenges, solutions, and applications,” *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826–3839, 2020.
- [28] A. Coates, A. Ng, and H. Lee, “An analysis of single-layer networks in unsupervised feature learning,” in *2011 International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 215–223, Ft. Lauderdale, FL, USA, 2011.
- [29] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: an evaluation of the state of the art,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.
- [30] P. Sun, H. Kretschmar, X. Dotiwalla et al., “Scalability in perception for autonomous driving: Waymo Open Dataset,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2443–2451, Seattle, WA, USA, 2020.
- [31] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, “Scalable person re-identification: a benchmark,” in *IEEE International Conference on Computer Vision (ICCV)*, pp. 1116–1124, Santiago, Chile, 2015.
- [32] J. Chen, C. Guo, S. Wei, X. Sun, and C. Feng, “SCO: a dataset for semantic communications,” <https://github.com/WeiSY0516/SCO-dataset.git>.
- [33] F. S. Melo, “Convergence of Q-learning: a simple proof,” *Institute of Systems and Robotics Technical Report*, pp. 1–4, 2001.