Joint optimization of computation offloading and resource allocation considering task prioritization in ISAC-assisted vehicular network

Article

Accepted Version

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3360962

Joint Optimization of Computation Offloading and Resource Allocation Considering Task Prioritization in ISAC-Assisted Vehicular Network

Dun Cao, *Member*, *IEEE*, Meihua Wu, Ning Gu, R. Simon Sherratt, *Fellow*, *IEEE*, Uttam Ghosh, *Senior Member*, *IEEE*, and Pradip Kumar Sharma, *Senior Member*, *IEEE*

***Abstract*—In the vehicular networks (VN) assisted by the integration of sensing and communication (ISAC), rapid processing of data from sensors is a necessary condition to ensure safe driving and enhance user experience. Utilizing the computational resources of the roadside unit (RSU) can effectively reduce the task processing delay. However, in some areas of the road, uneven distribution of task-vehicles can lead to severe load imbalance in neighbouring RSUs, and these tasks often have different delay requirements. The tasks in the high-load area can be offloaded to the low-load area to balance the load. We use the idle-vehicles in the low-load RSU area that are close to the task-vehicles as relays to hop and offload the tasks to the low-load RSUs. On the other hand, in order to satisfy the delay requirements of the heterogeneous tasks, this paper proposes the priority ordering of the heterogeneous tasks, the more delay-sensitive tasks require more resources to meet their delay requirements, i.e., the higher the priority. In order to both satisfy the delay requirements of heterogeneous tasks and maintain a small average system delay, we establish the optimization problem of minimizing the weighted average system delay and solve it by using the Relay Hopping and Differentiated Task Prioritization (RHATP) algorithm. Simulation results show that under the condition of guaranteeing the delay requirement of high-priority tasks, the strategy can achieve lower system delay and effectively reduce the processing delay in high-load areas. And it still maintains stable performance in different scenarios.**

***Index Terms*—ISAC, Edge Computing, VN, Task Prioritization.**

This work was supported by the National Natural Science Foundation of China (No.62272063, No.62072056 and No.61902041), the Natural Science Foundation of Hunan Province (No.2022JJ30617 and No.2020JJ2029), Standardization Project of Transportation Department of Hunan Province (B202108), Hunan Provincial Key Research and Development Program (2022GK2019), the Scientific Research Fund of Hunan Provincial Transportation Department (No.202042), the Open Project of Xiangjiang Laboratory (No.23XJ03003) and the Research Foundation of Education Bureau of Hunan Province, China(No.23A0253). *(Corresponding author: Pradip Kumar Sharma.)*

Dun Cao is an Associate Professor in the School of Computer and Communication Engineering at Changsha University of Science and Technology, Changsha, Hunan 410076, China. And she is the member of Xiangjiang Laboratory, Changsha 410205, China (e-mail: caodun@csust.edu.cn).

Meihua Wu is currently pursuing the M.S. degree in Changsha University of Science and Technology, Changsha, Hunan 410076, China (e-mail: wmh@stu.csust.edu.cn).

Ning Gu is currently pursuing the M.S. degree in Changsha University of Science and Technology, Changsha, Hunan 410076, China (e-mail: guning@stu.csust.edu.cn).

R. Simon Sherratt is currently with the School of Biomedical Engineering, the University of Reading, Reading, Berkshire RG6 6AH, UK (e-mail: r.s.sherratt@reading.ac.uk).

Uttam Ghosh is currently with the Department of Computer Science and Data Science, School of Applied Computational Sciences, Meharry Medical College, Nashville, TN 37208, USA (e-mail: ughosh@mmc.edu).

Pradip Kumar Sharma is currently with Department of Computing Science, University of Aberdeen, King's College, Aberdeen AB24 3FX, UK (e-mail: pradip.sharma@abdn.ac.uk).

I. Introduction

In the future sixth-generation (6G) network, the integration of sensing and communication (ISAC) is considered as one of the key technologies to support various new wireless services [1]. To improve the performance of ISAC systems, techniques such as reconfigurable intelligent surfaces (RIS) have been proposed [2], [3]. In vehicular networks, vehicles are usually equipped with typical sensors such as cameras, radar, lidar, and differential acoustic wave sensors [4]. The average number of sensors on a vehicle today is around 200 [5]. In order to more accurately recognize the information around the vehicle, the information collected by different sensors needs to be fused and processed [6], [14]. However, due to societal demands for the safe driving and the quality of service (QoS), vehicles need to compute this information faster, especially in certain unexpected situations. This is challenging for devices due to the limited local computing resources of vehicles. To solve this problem, mobile edge computing (MEC) technology is introduced. MEC pushes computing resources and services to the edge of the network, brings them closer to the user than the cloud [7], [8]. This reduces transmission delay and provides computing services to users, relieving pressure on local computing resources. By combining ISAC with MEC, vehicles can compute and respond to various environmental changes faster, improving safety and ensuring QoS [9–11].

There are many RSUs on the road, and the vehicles in their service area may generate many indivisible and heterogeneous tasks at the same time. The uneven distribution of task vehicles can lead to different loads for neighbouring RSUs. The computing and communicating resources of the high-load RSU will be tight, while the low-load RSU has a large amount of resources that are not fully utilized.

In order to reduce the load of high-load RSUs, we can make use of the computing resources of neighbouring RSUs and idle-vehicles. Since the computing power of RSUs is much stronger than that of vehicles, offloading these indivisible tasks to neighbouring RSUs is a better option than offloading them to idle-vehicles. Moreover, a large number of tasks uploaded at the same time tightens the communication resources of high-load RSUs. For task-vehicles at the edge of the communication range of high-load RSUs, long transmission distances result in degraded channel quality and reduced transmission rates. Offloading such tasks to neighbouring RSUs by hopping through relays (idle-vehicles close to the task-vehicles in the low-load RSUs) has a smaller transmission delay compared to transmitting over a link between RSUs, which is due to the small transmission delay caused by a large number of idle resources in the low-load RSUs.

In order to satisfy the different requirements of vehicle heterogeneous tasks on delay, and meanwhile to ensure the overall QoS of the system, assigning different priorities to heterogeneous tasks and appropriately tilting resources to high-priority tasks is a solution.

To address the aforementioned challenges, this paper focuses on the following aspects:

- In this paper, under the conditions of different task delay sensitivities and tight communication resources in the task-intensive area, we construct a multi-node cooperative model of task-vehicles, relay-vehicles and RSUs to jointly compute the offloading and resource allocation, and in this way, we construct a weighted average delay minimization problem.
- The proposed problem is a high-dimensional nonlinear problem and is mutually coupled and nonconvex. We decouple the problem into two subproblems: the unloading strategy problem and the resource allocation problem. Then we design a Relay Hopping and differentiated Task Prioritisation (RHATP) algorithm that combines one-to-one bilateral matching with convex optimization containing mixed integer constraints.
- In this paper, we adopt the marriage matching model in game theory to select out the task-vehicles and idle-vehicles that can form matching pairs. And after making the offloading decision, the problem will be transformed into a Mixed Integer Disciplined Convex Programming (MIDCP) problem with integer constraints, which can be utilized in convex optimization to perform the allocation of communication and computational resources, and to find an optimal solution to the problem.

The rest of this paper is organized as follows. The related works are reviewed in Section II. Section III describes the system model and expatiates on the research problems. In Section IV, we introduce the optimization approach in detail. Finally, performance evaluation and discussion are given in Section V. This work is concluded in Section VI.

II. Related Work

In order to reduce the delay of task processing, a large number of scholars have made many efforts.

Y. Liu *et al.* [13] addressed the cost trade-off between sensing and communication performance in ISAC systems, and proposed a generalized ds point-to-point ISAC model to account for scenarios in which the sensing state is different but correlated with the channel state, which supports the efficient application of ISAC in in-vehicle networks. However, the improvement not approach for scenarios with large differences in communication loads of neighboring RSUs. Y. Liu *et al.* [14] has designed a joint computation and resource allocation scheme for multi-RSU scenarios, taking into account the long-term system performance. This improves the energy efficiency of the fusion computing tasks generated at the vehicle side while guaranteeing the performance of the task-vehicle delays. However, the scheme does not consider the multi-RSU load imbalance and does not consider the priority of the tasks.

X. Li *et al.* [15] focus on the resource allocation problem of hybrid centralized/distributed V2X communication systems under different network load conditions. Vacanon resource blocks and power allocation algorithms are proposed for low-load networks, while for high-load networks, occupied resource blocks and power allocation algorithms are proposed with the main objective of maximizing the overall information value of V2X communication without exceeding the maximum user transmit power. X. He *et al.* [16] present a spectrum sharing problem in a scenario where vehicles move at high speeds and are uniformly distributed. They proposed a fingerprint-based deep Q-network suitable for distributed implementation by utilizing the fact that the V2V link will reuse the spectrum of the V2I link and that the V2V link acts as a proxy. The total capacity of V2I links and the payload delivery rate of V2V links are successfully improved. F. Li *et al.* [17] on the other hand proposed a V2X collaborative caching and resource allocation framework in a multi-RSU scenario. The framework enhances the edge caching capability through RSUs with surrounding idle-vehicle resources to avoid duplicate transmissions and reduce content access delay. Vehicles will cache requests to the MEC as well as nearby vehicles, and the article improves channel utilization through channel V2V multiplexing V2I.

Zubair Sharif *et al.* [18] mainly focus on the different delay requirements of heterogeneous tasks in edge computing (EC), set different tolerance delays for different types of tasks, and allocate the resources of edge nodes (ENs) according to the tolerance delay setting priority, so that tasks with different priorities can finally satisfy their tolerance delays. H. Zhao *et al.* [9] equip multiple MEC servers under a single RSU, and each MEC server is responsible for different types of tasks. Tasks uploaded to the RSU are set to different priorities based on the cutoff delay, and then dispatched by the RSU to different MEC servers for processing. The paper proposes a content-aware classification offloading algorithm for the balance relationship between delay and energy loss, which improves the message processing delay and energy loss while considering the task priority. Ruyuan Wang *et al.* [19] propose the average delay minimization and maximum individual delay minimization problems in a single base station (BS) scenario to address the differences in delay sensitivity of different tasks and use reinforcement learning to solve the problems and reduce the V2V link delay under the premise of ensuring the V2I link delay requirements, i.e., to improve the global network performance and ensure the fairness of a single user.

A long-term problem is considered in a multi-RSU load balancing scenario by Junhui Zhao *et al.* [20]. That is, utilizing vehicle local and RSU resources to achieve near-optimal delay

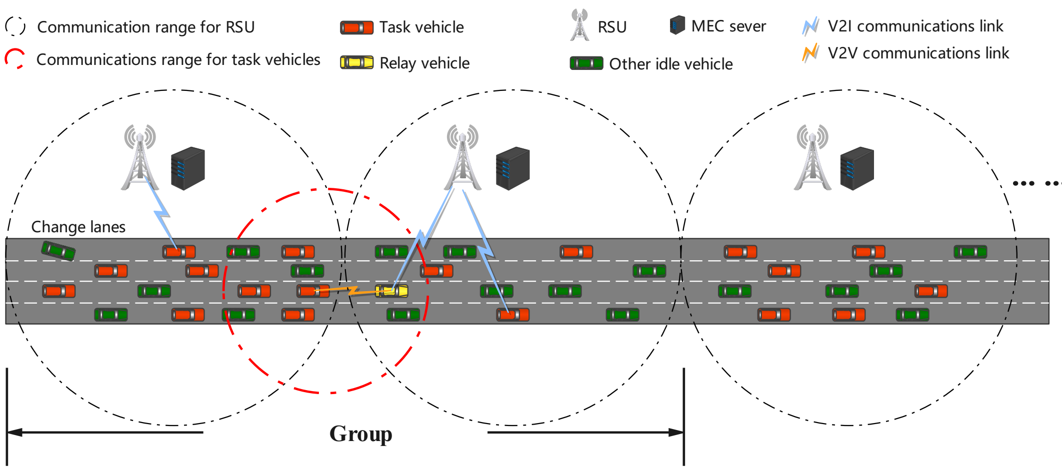


Fig. 1: ISAC-Assisted vehicular edge computing network considering prioritization

performance while satisfying time-averaged cost by integrating service caching, offloading strategies, resource allocation, and economic issues arising from energy consumption over time to satisfy time-averaged cost budget constraint performance. While Liu Lei *et al.* [21] then achieve computational load balancing in multi-RSU scenarios by transmitting tasks from computationally overloaded RSUs to nearby RSUs by links between RSUs.

In summary, existing work has explored RSUs under different communication loads, and optimization of neighbouring RSUs with different computational loads, but lacks research in the case where the communication loads of neighbouring RSUs also differ significantly. Moreover, only some of the above research papers explore the delay requirements of heterogeneous tasks in their own scenarios. In this paper, to address this situation, we use relay-vehicles to offload tasks to neighboring RSUs under the condition of guaranteeing the delay requirements of tasks with different priorities, in order to improve the spectrum utilization and reduce the system delay.

III. System Model

In this paper, we consider straight traveling roads. Since the vehicles traveling in the reverse direction have a large reverse relative speed for the task-vehicle, resulting in a shorter time to stay within the communication range of the task-vehicle. Therefore we simplify the scenario to a unidirectional straight road.

A. Task Model

In practical applications, the ISAC-assisted vehicle networking system realizes real-time information exchange of the whole vehicle network through advanced control methods and V2X links, and the RSU is the control center of the system. As shown in Fig. 1, we consider an wireless vehicular networks based on ISAC. Assuming that N RSUs are uniformly distributed on one side of the roadway, these RSUs form the set $\mathcal{S} = \{s_1, s_2, ..., s_n, ..., s_N\}, n \in [1, N]$. Each RSU is equipped with a server with computational resources F_{RSU} and communication resources B_{RSU}. I vehicles moving freely on the road form the set $\mathcal{V} = \{v_1, v_2, ..., v_i, ..., v_I\}, i \in [1, I]$. The data generated by the sensors forms a pending task on the vehicle. Denote the priority of a task by $q, q \in [1, Q]$, where the smaller q is the higher the priority. Thus, $w_{i,q}$, $w_{i,q} \in \{0, 1\}$ is used to define whether or not v_i generates a task, where 0 means that v_i does not generate a task with priority q, 1 means that v_i generates a task with priority q, and $\sum_{q\in[1,Q]} w_{i,q} \in \{0, 1\}, i \in [1, I]$. All $w_{i,q}$ make up the set $\mathcal{W} = \{w_{i,q}\}_{I\times Q}, \forall i \in [1, I], \forall q \in [1, Q]$. A task can be modeled by a set of parameters (e_i, f_i), where e_i denotes the size of communication resources required by the task generated by v_i and f_i denotes the size of computational resources required by the task generated by v_i. The task takes a binary offload.

B. Communication Model

Since the scheme proposed in this paper is more effective when the load difference between two RSUs is large, a group of RSUs will be used as an example in the following for ease of description. Two RSUs with neighbouring service ranges along the direction of vehicle travel are selected, and these two RSUs need to satisfy:

- The two RSUs have different loads.
- The low-load RSU needs to be the next RSU in the direction of vehicle travel.

All vehicles within the service area of these two RSUs form the set $\mathcal{V}', \mathcal{V}' = \{v_1, v_2, ..., v_j, ..., v_J\}, \mathcal{V}' \subset \mathcal{V}$. In order to better distinguish between task-vehicles and idle-vehicles within these two RSU service areas, this paper uses $v_{j_{\text{task}}}, \forall v_{j_{\text{task}}} \in \mathcal{V}'$ for task-vehicles and $v_{j_{\text{idle}}}, \forall v_{j_{\text{idle}}} \in \mathcal{V}' - \{v_{j_{\text{task}}}\}$ for idle-vehicles. We select a task-vehicle in the high-load RSU to illustrate the task processing delay. The high-load RSU is the current RSU of the task-vehicle, defined as s_m, and the low-load RSU is the neighboring RSU, defined as $s_{m'}$, $\forall s_m \in \mathcal{S}, \forall s_{m'} \in \mathcal{S} - \{s_m\}$.

The driving trajectory prediction model is constructed by predicting the position information at the moment $t + \Delta t$ based on the position information at the moment t [22]. In this paper, all vehicles on the road are traveling in one direction, and the two-dimensional coordinates are established according to the parallel road in the direction of vehicle travel, and the two-dimensional coordinates of $v_j, \forall j \in [1, J]$ at the $t+\Delta t$ moment are:

$$\begin{cases} x_j^{t+\Delta t} = x_j^t + u_j^t \Delta t + \frac{1}{2}\frac{\mathrm{d}u_j^t}{\mathrm{d}t}(\Delta t)^2 \\ y_j^{t+\Delta t} = y_j^t \pm rand\{0,1\}W \end{cases}, j \in [1, J], \tag{1}$$

where $\frac{\mathrm{d}u_j^t}{\mathrm{d}t}$ denotes the acceleration of the v_j at the moment t; $rand\{0,1\}$ function is used to generate random 0 or 1, i.e., at most one lane change is allowed in the Δt time; W denotes the lane width. Then the euclidean distance between task-vehicle $v_{j_{\text{task}}}$ and the idle-vehicle $v_{j_{\text{idle}}}$ at the moment $t+\Delta t$ can be obtained as:

$$\begin{aligned} d_{j_{\text{task}},j_{\text{idle}}}^{t+\Delta t} = \sqrt{(x_{j_{\text{task}}}^{t+\Delta t} - x_{j_{\text{idle}}}^{t+\Delta t})^2 + (y_{j_{\text{task}}}^{t+\Delta t} - y_{j_{\text{idle}}}^{t+\Delta t})^2}, \\ \forall v_{j_{\text{task}}} \in \mathcal{V}', \forall v_{j_{\text{idle}}} \in \mathcal{V}' - \{v_{j_{\text{task}}}\}. \end{aligned} \tag{2}$$

Similarly, the euclidean distance between v_j and the current RSU s_m at the moment $t+\Delta t$ is:

$$d_{j,m}^{t+\Delta t} = \sqrt{(x_j^{t+\Delta t} - x_m^{t+\Delta t})^2 + (y_j^{t+\Delta t} - y_m^{t+\Delta t})^2}, \forall v_j \in \mathcal{V}', \forall s_m \in \mathcal{S}. \tag{3}$$

In this paper, orthogonal frequency division multiplexing (OFDM) is used for orthogonal subchannel assignment to reduce channel interference [21], [23]. And there is no channel multiplexing between V2V link and V2I link. Therefore channel interference can be ignored in channel modelling. The transmission rate between v_j and s_m is [24]:

$$r_{j,m} = b_{j,m}\frac{B_{\text{RSU}}}{X}\log_2(1 + \frac{p_j d_{j,m}^{-\gamma} h^2}{N_0}), \forall v_j \in \mathcal{V}', \forall s_m \in \mathcal{S}, \tag{4}$$

where X denotes that the communication resources of the RSU are divided into X parts and X is a positive integer [25], [26]; $b_{j,m}$ denotes the number of channel allocated to vehicle j, which is a positive integer, and $\sum_{j \in [1,J]} b_{j,m} = X$, all $b_{j,m}$ form the set $\boldsymbol{b}$; p_j denotes the transmit power of v_j; h^2 is channel gain, and γ path loss index. Since V2V uses still the communication resources of the current RSU if the task is offloaded to the neighboring RSU by the relay-vehicle. Therefore, the communication resources shared by $v_{j_{\text{task}}}$ and $v_{j_{\text{idle}}}$ are still denoted by $b_{j_{\text{task}},m}$. The transmission rate between $v_{j_{\text{task}}}$ and $v_{j_{\text{idle}}}$ is [21]:

$$r_{j_{\text{task}},j_{\text{idle}}} = b_{j_{\text{task}},m}\frac{B_{\text{RSU}}}{X}\log_2(1 + \frac{p_{j_{\text{task}}} d_{j_{\text{task}},j_{\text{idle}}}^{-\gamma} h^2}{N_0}). \tag{5}$$

C. Computational Model

In this paper, the return delay is ignored because the vehicle return from the RSU is a small processed data [22]. By comparing the delay between a task-vehicle offloading a task to the current RSU and using a relay-vehicle to offload the task to the next RSU (neighboring), the task-vehicle has the option of offloading the task to the current RSU or to a neighboring RSU. $\alpha_{j_{\text{task}},m}$ denotes that the task is offloaded from $v_{j_{\text{task}}}$ to the current RSU s_m. $\alpha_{j_{\text{task}},m} = 0$ or 1, 0 is for no offloading to s_m, and 1 is for offloading to s_m, $0 \le \sum_{m \in [1,N]} \alpha_{j_{\text{task}},m} \le 1$. Similarly, $\alpha_{j_{\text{task}},m'}$ denotes that the task is offloaded from $v_{j_{\text{task}}}$ to $s_{m'}$, all $\alpha_{j_{\text{task}}}$ form the set $\boldsymbol{\alpha}$. Use $R_{j_{\text{task}}}$, R_m, and $R_{m'}$ to denote the service ranges of $v_{j_{\text{task}}}$, s_m, $s_{m'}$, respectively. When the vehicle trajectory satisfies the conditions described by Eq. (6), The transmission delay can be expressed as Eq. (7).

$$\begin{cases} |d_{j_{\text{task}},j_{\text{idle}}}^{t+\Delta t}| \le R_{j_{\text{task}}}, |d_{j_{\text{task}},m}^{t+\Delta t}| \le R_m, |d_{j_{\text{idle}},m'}^{t+\Delta t}| \le R_{m'}, \\ \forall v_{j_{\text{task}}} \in V', \forall v_{j_{\text{idle}}} \in V' - \left\{v_{j_{\text{task}}}\right\}, \forall s_m \in S, \forall s_{m'} \in S - \{s_m\}, \\ \forall \Delta t \in [0, \max(D_{j_{\text{task}}}^{\text{local}}, D_{j_{\text{task}}}^{\text{com}} + D_{j_{\text{task}}}^{\text{off}})]. \end{cases} \tag{6}$$

$$\begin{aligned} D_{j_{\text{task}}}^{\text{com}} &= \alpha_{j_{\text{task}},m} D_{j_{\text{task}}}^{\text{current}} + \alpha_{j_{\text{task}},m'} D_{j_{\text{task}}}^{\text{neighbor}} \\ &= w_{j_{\text{task}},q}\left[\alpha_{j_{\text{task}},m}\frac{e_{j_{\text{task}}}}{r_{j_{\text{task}},m}} + \alpha_{j_{\text{task}},m'}\left(\frac{e_{j_{\text{task}}}}{r_{j_{\text{task}},j_{\text{idle}}}} + \frac{e_{j_{\text{task}}}}{r_{j_{\text{idle}},m'}}\right)\right], \end{aligned} \tag{7}$$

where $\frac{e_{j_{\text{task}}}}{r_{j_{\text{task}},m}}$ denotes the transmission delay while the task is offloaded onto the current RSU s_m and $\frac{e_{j_{\text{task}}}}{r_{j_{\text{task}},j_{\text{idle}}}} + \frac{e_{j_{\text{task}}}}{r_{j_{\text{idle}},m'}}$ denotes the transmission delay while it is offloaded onto the neighbouring RSU s'_m.

Depending on whether the task is unloaded or not, the computational delay of the task-vehicle can be divided into 2 types:

1) ***Tasks are directly processed locally in the vehicle without offloading.*** The local computational delay can be denoted as:

$$D_{j_{\text{task}}}^{\text{local}} = w_{j_{\text{task}},q}\frac{f_{j_{\text{task}}}}{F_{j_{\text{task}}}}, j_{\text{task}} \in [1, J], m \in [1, N], \tag{8}$$

where $f_{j_{\text{task}}}$ denotes the number of CUP cycles required for the task of task-vehicle $v_{j_{\text{task}}}$, and $F_{j_{\text{task}}}$ denotes the size of the computational resource of $v_{j_{\text{task}}}$.

2) ***The task is offloaded to the current RSU or neighbouring RSUs.*** i.e. The computational delay can be expressed as:

$$D_{j_{\text{task}}}^{\text{exe}} = w_{j_{\text{task}},q}\frac{f_{j_{\text{task}}}}{\sum_{m \in [1,N]} c_{j_{\text{task}},m}\alpha_{j_{\text{task}},m}F_{\text{RSU}}}, \tag{9}$$

where $c_{j_{\text{task}}}$ denotes the ratio factor of computing resources allocated by the RSU for the tasks of $v_{j_{\text{task}}}$, and all $c_{j_{\text{task}}}$ form the set $\boldsymbol{c}$. Total delay of $v_{j_{\text{task}}}$ is:

$$\begin{aligned} D_{j_{\text{task}}} &= (1 - \sum_{m \in [1,N]} \alpha_{j_{\text{task}},m}) D_{j_{\text{task}}}^{\text{local}} + \sum_{m \in [1,N]} \alpha_{j_{\text{task}},m} D_{j_{\text{task}}}^{\text{off}} \\ &= (1 - \sum_{m \in [1,N]} \alpha_{j_{\text{task}},m}) D_{j_{\text{task}}}^{\text{local}} + \sum_{m \in [1,N]} \alpha_{j_{\text{task}},m}(D_{j_{\text{task}}}^{\text{com}} + D_{j_{\text{task}}}^{\text{exe}}). \end{aligned} \tag{10}$$

For RSUs on the road that do not meet the screening criteria, the vehicles in their service range will process the tasks locally or offload them to the current RSUs, and Eq. (9) also calculates the task processing delay, so Eq. (9) can be extended to the whole road. Meanwhile, considering the priority of different tasks and the goal of minimizing the average system delay, this paper weights the delay of different priority tasks. In summary, the problem of this paper can be expressed as

$$\begin{aligned}
&\mathbf{P1:} \min_{\boldsymbol{b},\boldsymbol{\alpha},\boldsymbol{c}} \overline{D} = \min_{\boldsymbol{b},\boldsymbol{\alpha},\boldsymbol{c}} \tfrac{1}{V^t} \sum_{i\in[1,I]} \sum_{q\in[1,Q]} \eta_q w_{i,q} D_i \\
&s.t.\ \mathrm{C}_1: \sum_{q\in[1,Q]} \eta_q = 1, 1 > \eta_1 > \eta_2 > ... > \eta_q > ... > \eta_Q > 0, \\
&\quad \mathrm{C}_2: \sum_{q\in[1,Q]} w_{i,q} \in \{0,1\}, w_{i,q} \in \{0,1\}, \forall i \in [1,I], \\
&\quad \mathrm{C}_3: |d_{i,n}^{t+\Delta t}| \le R_n, |d_{i_{\mathrm{relay}},n'}^{t+\Delta t}| \le R_{n'}, |d_{i,i_{\mathrm{relay}}}^{t+\Delta t}| \le R_i, \forall v_i, v_{i_{\mathrm{relay}}} \in \mathcal{V}, \\
&\qquad \forall s_n, s_{n'} \in \mathcal{S}, \forall \Delta t \in [0, \max(D_i^{\mathrm{local}}, D_i^{\mathrm{com}} + D_i^{\mathrm{off}})], \\
&\quad \mathrm{C}_4: 0 \le b_{i,n} \le X, \sum_{v_i\in\mathcal{V}} b_{i,n} = X, \\
&\quad \mathrm{C}_5: 0 \le c_{i,n} \le 1, \sum_{v_i\in\mathcal{V}} \alpha_{i,n} c_{i,n} = 1, \forall v_i \in \mathcal{V}, \\
&\quad \mathrm{C}_6: 0 \le \sum_{s_n\in\mathcal{S}} \alpha_{i,n} \le 1, \alpha_{i,n} \in \{0,1\},
\end{aligned} \tag{11}$$

where C_1 guarantees that the higher the priority the higher the weight, where η_q defines the weight with priority q. C_2 guarantees that a vehicle generates at most one task at a time and that this task can only be of one priority. C_3 guarantees that during the time that a task is being processed, the task-vehicle is within the communication range of the current RSU, the relay-vehicle is within the communication range of the neighbouring RSU, and the relay-vehicle is within the communication range of the task-vehicle. C_4 guarantees that the sum of the bandwidth shares that each task-vehicle is allocated is equal to the bandwidth shares owned by the RSU. C_5 ensures that the ratio factor of computational resources allocated to each task-vehicle lies in the range of 0 to 1, and the sum of the ratios within the communication range of one RSU is equal to 1. C_6 ensures that the tasks are processed locally or at the current RSU or at the neighbouring RSUs. V^t denotes the total number of vehicles generating the tasks at the moment t.

IV. Algorithm Design

Since the problem contains integer variables $\boldsymbol{b}$ and $\boldsymbol{\alpha}$, P1 is a mixed integer nonconvex optimization problem, which is an NP-hard problem. In order to solve P1, this paper proposes a RHATP algorithm, in which a game-theory based offloading strategy and a convex optimisation based resource allocation method are proposed respectively, where the former determines the set of offloading strategies $\boldsymbol{\alpha}$ for the computational tasks and the latter determines the set of resource allocations $\boldsymbol{b}$, $\boldsymbol{c}$ for the RSUs.

A. Unloading Mechanism Based on Game Theory

This section determines the offloading strategy of the task-vehicles, all of which do variable speed linear motion. In this paper, the game-theoretic repetitive culling strictly inferior strategy is used to obtain the set of vehicles $\mathcal{V}^{\mathrm{idle}}$, $\mathcal{V}^{\mathrm{idle}} \subset \mathcal{V}$ that can be used as relay nodes according to the constraints C_3. The task is selected to be offloaded only when the task offloading delay is less than the local computation delay, so Δt will be valued as:

$$\Delta t = \frac{f_i}{F_i}. \tag{12}$$

Vehicles acting as relay nodes need to fulfill the requirement of being within the service range of the task-vehicle during the Δt time.

Setting initial values for sets $\boldsymbol{b}$ and $\boldsymbol{c}$ that satisfy constraints C_4, C_5, then P1 is transformed into

$$\begin{aligned}
&\mathbf{P2:} \min_{\alpha} \overline{D} = \min_{\alpha} \tfrac{1}{V^t} \sum_{i\in[1,I]} \sum_{q\in[1,Q]} \eta_q w_{i,q} D_i \\
&s.t.\ \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \mathrm{C}_6.
\end{aligned} \tag{13}$$

We denote the task-vehicles by $v_{i_{\mathrm{task}}}$ and compose the set $\mathcal{V}^{\mathrm{task}}, \mathcal{V}^{\mathrm{task}} \subset \mathcal{V}$. In the situation that the vehicles satisfy constraint C3, the optional idle-vehicles $v_{i_{\mathrm{idle}}}$ of all task-vehicles together form the set $\mathcal{V}^{\mathrm{idle}}, \mathcal{V}^{\mathrm{idle}} \subset \mathcal{V}$. The set of task-vehicles $\mathcal{V}^{\mathrm{task}}$ and the set of optional idle-vehicles $\mathcal{V}^{\mathrm{idle}}$ can be viewed as two sets of disjointed selfish and rational participants, and P2 can be viewed as a one-to-one bilateral matching between $\mathcal{V}^{\mathrm{task}}$ and $\mathcal{V}^{\mathrm{idle}}$. Therefore, the bilateral matching marriage model in game theory can be used to formulate the matching strategy. Here $(i_{\mathrm{task}}, i_{\mathrm{idle}}), \forall i_{\mathrm{task}}, i_{\mathrm{idle}} \in [1, I]$ is used to denote the matching pair, and the one-to-one matching of the network is defined as Υ, which satisfies the following conditions: 1) $\Upsilon(i_{\mathrm{idle}}) \in \mathcal{V}^{\mathrm{task}}, \forall i_{\mathrm{idle}} \in \mathcal{V}^{\mathrm{idle}}$; 2) $\Upsilon(i_{\mathrm{idle}}) = i_{\mathrm{task}} \Leftrightarrow \Upsilon(i_{\mathrm{task}}) = i_{\mathrm{idle}}$. With the matching Υ, it can be derived as:

$$\alpha_{i_{\mathrm{task}},n'} = \begin{cases} 1, & \text{if } \Upsilon(i_{\mathrm{task}}) = i_{\mathrm{idle}}, \\ 0, & \text{else.} \end{cases} \tag{14}$$

In order to describe the performance of each participant in the matching, we set a preference for both parties. For each vehicle, a preference can be set only if the other vehicle is within one's communication range and belongs to the set $\mathcal{V}^{\mathrm{idle}}$, otherwise the other vehicle is regarded as unselectable. For the task-vehicle, its preference for the idle-vehicle can be denoted as:

$$U_{i_{\mathrm{task}}}(i_{\mathrm{idle}}) = \frac{e_{i_{\mathrm{task}}}}{r_{i_{\mathrm{task}},i_{\mathrm{idle}}}} + \frac{e_{i_{\mathrm{task}}}}{r_{i_{\mathrm{idle}},n'}} + \frac{f_{i_{\mathrm{task}}}}{c_{i_{\mathrm{task}},n'} F_{\mathrm{RSU}}}. \tag{15}$$

Similarly, for the optional idle-vehicle $v_{i_{\mathrm{idle}}}$, it has a preference $U_{i_{\mathrm{idle}}}(i_{\mathrm{task}}) = U_{i_{\mathrm{task}}}(i_{\mathrm{idle}})$ for the task-vehicle $v_{i_{\mathrm{task}}}$.

To describe each participant's preference for both parties, a matching preference relation $(\succ_{i_{\mathrm{task}}}, \succ_{i_{\mathrm{idle}}})$ is introduced here. If task-vehicle prefers i_{idle} to i'_{idle}, there is:

$$\begin{aligned}
&i_{\mathrm{idle}} \succ_{i_{\mathrm{task}}} i'_{\mathrm{idle}} \Leftrightarrow U_{i_{\mathrm{task}}}(i_{\mathrm{idle}}) < U_{i_{\mathrm{task}}}(i'_{\mathrm{idle}}), \\
&\forall i_{\mathrm{task}}, i_{\mathrm{idle}}, i'_{\mathrm{idle}} \in [1, I], i_{\mathrm{task}} \ne i_{\mathrm{idle}} \ne i'_{\mathrm{idle}}.
\end{aligned} \tag{16}$$

If optional idle-vehicle $v_{i_{\mathrm{idle}}}$ prefers i_{task} to i'_{task}, there is:

$$\begin{aligned}
&i_{\mathrm{task}} \succ_{i_{\mathrm{idle}}} i'_{\mathrm{task}} \Leftrightarrow \begin{cases} q < q' \\ q = q', \mathit{and}\ U_{i_{\mathrm{idle}}}(i_{\mathrm{task}}) < U_{i_{\mathrm{idle}}}(i'_{\mathrm{task}}), \end{cases} \\
&\forall i_{\mathrm{task}}, i'_{\mathrm{task}}, i_{\mathrm{idle}} \in [1, I], \forall q, q' \in [1, Q], i_{\mathrm{task}} \ne i'_{\mathrm{task}}.
\end{aligned} \tag{17}$$

where q and q' denote the task priority of i_{task} and i'_{task}, respectively.

Based on the matching preferences, the preference lists of the two participants are created and the two parties are matched based on the preference lists. Step 1: All task-vehicles send requests to their most preferred idle-vehicles. The idle-vehicle that receives the request will keep the request from the most preferred task-vehicle and rejects the requests from the

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3360962

other vehicles, and the vehicles that are matched successfully will change to the matched status. Step 2: All rejected task-vehicles will send requests to the most preferred of the idle-vehicles that have not yet rejected them (if all optional idles have already rejected the task-vehicle, it will not send another request). The above steps are repeated until all optional idle-vehicles have been matched or no task-vehicle sends a request, at which point the algorithm will reach a steady state. The following is the proof of the stability of the Algorithm 1.

Proof: Assume that the bilateral matching marriage model algorithmic matching does not include pairwise $(i_{\text{task}}, i_{\text{idle}})$. Then two situations will exist:

- Task i_{task} never initiates application to i_{idle}. Since i_{task} initiates the application in descending order, i_{task} prefers the match obtained by Algorithm 1 to i_{idle}. So $(i_{\text{task}}, i_{\text{idle}})$ is not unstable.
- i_{task} applies to i_{idle}. So i_{idle} rejects i_{task}, i_{idle} prefers the match the algorithm gets $(i_{\text{task}}, i_{\text{idle}})$ is not a destabilising factor.

Therefore, the Algorithm 1 is not unstable, it is stable and can reach the Nash equilibrium. The proof is complete. According to the above analysis, the complexity of Algorithm 1 is mainly determined by the number of vehicles of both parties involved in matching, so the complexity of Algorithm 1 is $O(I^2)$.

After getting the $\mathcal{V}_{\text{cp}}^{\text{task}}$, if $\mathcal{V}_{\text{cp}}^{\text{task}}[i_{\text{task}}] \neq 0$, then $\alpha[i_{\text{task}}, n'] = 1$, where n' is the seral number of neighbour RSU. For all unmatched task-vehicles, $\alpha[i_{\text{task}}, n'] = 0$, and to idle-vehicles, $\alpha[i_{\text{idle}}, n']$ are uniformly set to 0.

Algorithm 1 Bilateral Matching Marriage Model

Input:
$\mathcal{V}, U_{i_{\text{task}}}, U_{i_{\text{idle}}}$.
Output:
$\mathcal{V}_{\text{cp}}^{\text{task}}$.

```
Initialization: The set of unmatched task-vehicles V^task, the set of unmatched relay-vehicles V^idle, and the set of task-vehicle matching pairs V_cp^task = 0.
The matching preferences of the task-vehicle and the relay-vehicle are sorted according to U_{i_task}(i_idle), U_{i_idle}(i_task) and task priority respectively to obtain the matching preference sequences ≻_{i_task} and ≻_{i_idle};
while V^task ≠ ∅ and V^idle ≠ ∅ do
    count=0;
    for each i_task in [1, I] do
        if V_cp^task[i_task] = 0 and ≻_{i_task} exists acceptable relay-vehicles then
            Send a request to the first vehicle i_idle of ≻_{i_task},count=1;
        end if
    end for
    if count==0 then
        No task-vehicle has sent a request, break;
    end if
    for each i_idle in [1, I] do
        if i_idle receives a request (with or without a match) then
            for each i_task in [1, I] do
                if i_task is i_idle's preferred task-vehicle then
                    V_cp^task[i_task] = i_idle;
                    V^task[i_task] = 0;
                    V^idle[i_idle] = 0;
                end if
            end for
            if i_idle < I − 1 then
                Discard requests for unmatched vehicles, the task-vehicle for which the request is discarded: V^task[n] = 1;
            end if
        end if
    end for
end while
return V_cp^task
```

B. Resource Allocation Algorithm Based on Convex Optimization

Based on the $\boldsymbol{\alpha}$ obtained from Algorithm 1, P2 is transformed into

$$
\begin{aligned}
&\mathbf{P3:} \min_{\boldsymbol{b},\boldsymbol{c}} \overline{D} = \min_{\boldsymbol{b},\boldsymbol{c}} \frac{1}{V'} \sum_{i\in[1,I]} \sum_{q\in[1,Q]} \eta_q w_{i,q} D_i \\
&s.t.\ \mathrm{C}_1,\mathrm{C}_2,\mathrm{C}_3,\mathrm{C}_4,\mathrm{C}_5.
\end{aligned} \tag{18}
$$

The objective function is a convex problem [20] and its proof follows.

Proof : Let

$$
\begin{cases}
SN_{i_{\text{task}},n} = \frac{B_{\text{RSU}}}{X} \log_2\left(1 + \frac{p_{i_{\text{task}}} d_{i_{\text{task}},n}^{-\gamma} h^2}{N_0}\right), \\
SN_{i_{\text{task}},i_{\text{idle}}} = \frac{B_{\text{RSU}}}{X} \log_2\left(1 + \frac{p_{i_{\text{task}}} d_{i_{\text{task}},i_{\text{idle}}}^{-\gamma} h^2}{N_0}\right), \\
SN_{i_{\text{idle}},n'} = \frac{B_{\text{RSU}}}{X} \log_2\left(1 + \frac{p_{i_{\text{task}}} d_{i_{\text{idle}},n'}^{-\gamma} h^2}{N_0}\right),
\end{cases} \tag{19}
$$

then the transmission delay of the task can be expressed as:

$$
\Gamma(\boldsymbol{b}) = \sum_{i_{\text{task}}\in[1,I]} \sum_{q\in[1,Q]} \eta_q w_{i_{\text{task}},q} \left[\alpha_{i_{\text{task}},n} \frac{e_{i_{\text{task}}}}{b_{i_{\text{task}},n} SN_{i_{\text{task}},n}} + \alpha_{i_{\text{task}},n'} \left(\frac{e_{i_{\text{task}}}}{b_{i_{\text{task}},n} SN_{i_{\text{task}},n}} + \frac{e_{i_{\text{task}}}}{b_{i_{\text{idle}},n'} SN_{i_{\text{idle}},n'}}\right)\right]. \tag{20}
$$

And the computational delay is expressed as:

$$
\Phi(\boldsymbol{c}) = \sum_{i_{\text{task}}\in[1,I]} \sum_{q\in[1,Q]} \eta_q w_{i_{\text{task}},q} \left[\left(1 - \sum_{n\in[1,N]} \alpha_{i_{\text{task}},n}\right) \frac{f_{i_{\text{task}}}}{F_{i_{\text{task}}}} + \sum_{n\in[1,N]} \alpha_{i_{\text{task}},n} \frac{f_{i_{\text{task}}}}{\sum_{n\in[1,N]} c_{i_{\text{task}},n} \alpha_{i_{\text{task}},n} F_{\text{RSU}}}\right], \tag{21}
$$

then P3 is transformed into

$$
\begin{aligned}
&\mathbf{P4:} \min_{\boldsymbol{b},\boldsymbol{c}} \overline{D} = \min_{\boldsymbol{b},\boldsymbol{c}} \frac{1}{V'} (\Phi(\boldsymbol{c}) + \Gamma(\boldsymbol{b})) \\
&s.t.\ \mathrm{C}_1,\mathrm{C}_2,\mathrm{C}_3,\mathrm{C}_4,\mathrm{C}_5.
\end{aligned} \tag{22}
$$

When both $\Phi(\boldsymbol{c})$ and $\Gamma(\boldsymbol{b})$ are convex, the objective function is convex. For each pair of $(c_i, c_{i'}), \forall i, i' \in [1, I]$, the Hessian matrix of $\Phi(\boldsymbol{c})$ can be found as:

$$
\frac{\partial^2 \Phi}{\partial c_i \partial c_{i'}} = \begin{cases}
\sum_{i\in[1,I]} \sum_{q\in[1,Q]} \sum_{n\in[1,N]} \alpha_{i,n} \eta_q w_{i,q} \frac{2 f_i}{\sum_{n\in N} c_{i,n}^3 \alpha_{i,n} F_{\text{RSU}}}, & \text{if } i = i' \\
0, & \text{else.}
\end{cases} \tag{23}
$$

Algorithm 2 Convex Optimisation

Input:
$\boldsymbol{\alpha}$.
Output:
$\boldsymbol{b}, \boldsymbol{c}$.
Initialization: Setting the set of continuous variables $\boldsymbol{c}$ and the set of integer variables $\boldsymbol{b}$;
Write the corresponding optimisation objective and constraints according to P4;
Calling CLPEX package with CVXPY and solving it
return $\boldsymbol{b}, \boldsymbol{c}$

It can be seen that the matrix is a real symmetric matrix. Here this Hessian matrix is represented by the matrix H. Then setting a nonzero vector $A \in R^I$, it is easy to see that $A^T HA \geq 0$ holds permanently, so the matrix H is a semipositive definite matrix. Therefore, $\Phi(\boldsymbol{c})$ is a convex function, and similarly, $\Gamma(\boldsymbol{b})$ is also a convex function. In summary, the objective function is a convex function, and the proof is complete.

The problem can be solved using the CVXPY package. Due to the presence of integer constraints C_4 and non-integer constraints C_5 , P4 is a MIDCP problem [27], which can be solved optimally using CLPEX academic edition [28]. See Algorithm 2 for details. The complexity of CLPEX depends on the number of variables in the input, so the complexity of Algorithm 2 is $O((2 \cdot I \cdot N)^2)$.

Based on the $\boldsymbol{b}$ and $\boldsymbol{c}$ solved by Algorithm 2 , compare the offloading delay and the local computation delay, if the local delay of v_i is the smallest, then update $\alpha[i,:] = 0$. Repeatedly call Algorithm 1, Algorithm 2 until the average system delay change reaches the threshold ζ. See Algorithm 3 for details.

V. Simulation results and Performance analysis

In this section, we evaluate the performance of the proposed RHATP through simulations.

A. Simulation Setting

In this paper, a 2.5km long, 4-lane scenario is simulated. 4 RSUs are uniformly distributed on one side of the road, and the vehicles on the road obey the Poisson distribution. Tasks are divided into 2 priorities. The main parameter settings are shown in Table 1, some of these reference [18] and [20], with minor adjustments based on our scenario.

To illustrate the performance of the RHATP proposed in this paper, it will be compared with the following 2 strategies:

*1) **No Relay Hopping (NRH):*** No Relay Hopping (NRH): The task is divided into two priority levels. Tasks can only be computed locally in the vehicle or offloaded to the current RSU.

*2) **No Differentiation of Task Priorities (NTP):*** Tasks do not differentiate between priorities. Tasks can choose to be computed locally, offloaded to the current RSU for computation, or offloaded to a neighbouring RSU by hopping through an idle-avehicle.

Algorithm 3 RHATP

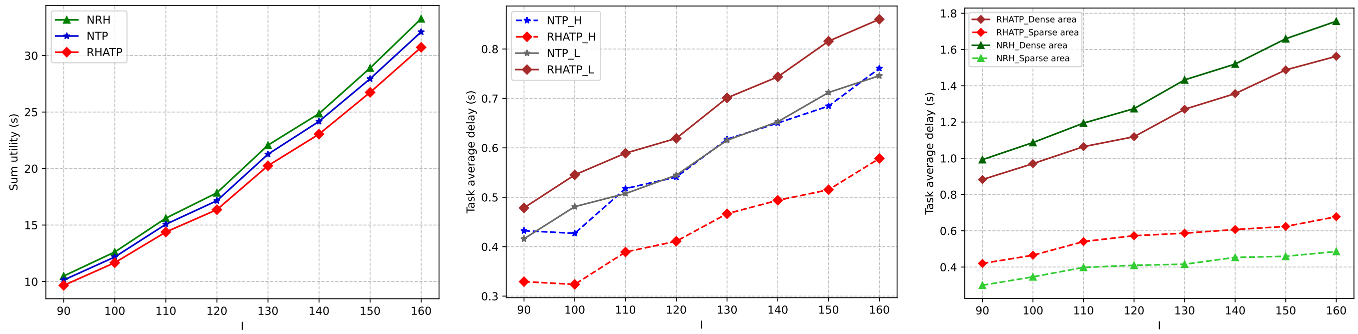
Input:
$\mathcal{V}, \mathcal{S}$.
Output:
$\boldsymbol{\alpha}, \boldsymbol{b}, \boldsymbol{c}$.
Initialization: G is infinite, threshold value ζ;
Setting initial values for sets $\boldsymbol{b}$ and $\boldsymbol{c}$ that satisfy constraints C_4, C_5;
$U_{i_{\text{task}}}$ and $U_{i_{\text{idle}}}$ are calculated according to Eq. (15);
Calling Algorithm 1 yields $\mathcal{V}_{\text{cp}}^{\text{task}}$;
Calling Algorithm 2 yields $\boldsymbol{b}, \boldsymbol{c}$;
Calculating the average system delay $\overline{D}_1$, set $\overline{D}_2 = 0$;
while $G > \zeta$ **do**
  for each i in $[1, I]$ **do**
    if $D_i^{\text{local}} \leq D_i^{\text{off}}$ **then**
      $\alpha[i,:] = 0$;
    end if
  end for
  Calling Algorithm 2 yields $\boldsymbol{b}, \boldsymbol{c}$;
  Calling Algorithm 1 yields $\mathcal{V}_{\text{cp}}^{\text{task}}$;
  for each i in $[1, I]$ **do**
    Find the RSU where i is located n;
    if $\mathcal{V}_{\text{cp}}^{\text{task}}[i] \neq 0$ **then**
      $\alpha[i,:] = 0$;
      $\alpha[i,n'] = 1$;
    end if
  end for
  Calculating the average system delay $\overline{D}_2$;
  $G = \overline{D}_2 - \overline{D}_1$;
end while
return $\boldsymbol{\alpha}, \boldsymbol{b}, \boldsymbol{c}$

B. Performance Evaluation

The sum utility is defined as $\sum_{i \in [1,I]} \sum_{q \in [1,Q]} D_i \eta_q w_{i,q}$, which is the weighted total system delay. The probability of occurrence

TABLE I: Key Symbols Throughout Simulation

Symbol	Definition	Value
p_i	Vehicle transmitting power	27 dBm
F_{RSU}	Computing resources available to RSUs	7000 MHz
B_{RSU}	Communication resources available to RSUs	110 Mb
F_i	Vehicle i computing resources	91 MHz
N_0	Noise power	-114 dBm
γ	Path loss exponent	2.3
X	Number of split channels	500
h	Channel gains	2×10^{-3}
f_i	Needed computation amount of i	[109,140] MHz
e_i	Workload size of i	[7,9] Mb
u_i^t	Velocity of i at time t	[15,30] m/s
$\frac{\text{d}u_i^t}{\text{dt}}$	Acceleration of i at time t	[-1,1] m/s^2
W	Single lane width	5 m
η_1	Weighting of high-priority tasks	0.7
ζ	Threshold	0.01



(a) Sum utility (b) Task average delay (c) Task average delay for different task load regions

Fig. 2: The effect of the number of vehicles I

of high-priority tasks is defined as P_H, and the ratio of the number of tasks in high-density and low density zones is defined as RAT.

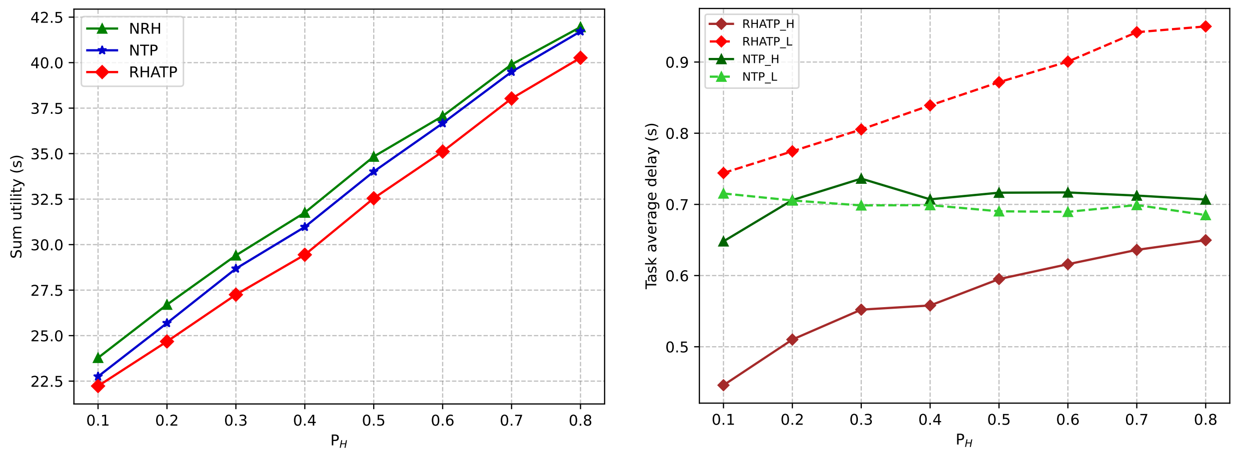
*1) **The effect of the number of vehicles I:*** P_H and RAT are set to 0.3 and 4, respectively. Fig. 2 (a) shows how the sum utilities of RHATP, NRH and NTP varies with the number of vehicles I. The sum utilities of all three strategies increase with I, with RHATP, the scheme proposed in this paper, has the smallest utility. Since NTP does not distinguish the priority of tasks, the task average delay of its high-priority tasks RHATP_H is significantly lower than the task average delay of low-priority tasks RHATP_L , while the task average delays of the two priorities of NTP are basically the same, as shown in Fig. 2 (b). Whereas the higher priority task delay is weighted more heavily, this leads to a larger sum utility for NTP. For NRH, it does not offload the tasks by hopping, so the average delay of the tasks in the high-load RSU region will be large, while the average delay of tasks in the low-load RSUs will be relatively small, but since the number of tasks in the high-load RSUs is much larger than that in the low-load RSUs, the total utility will still be relatively large, as shown in Fig. 2 (c). Since the number of pairable task-vehicles and idle-vehicles saturates at a certain level of I, the difference in sum utility between RHATP and NRH increases as I increases until saturation is reached and the difference no longer increases. In summary, Fig. 2 (a) and Fig. 2 (b) show that RHATP ensures and utility minimization while effectively reducing the processing delay of high-priority tasks. While Fig. 2 (a) and Fig. 2 (c) show that this paper effectively reduces the average delay of tasks in the high-load RSU service area through the hopping operation and in this way achieves the reduction of the sum utility.

*2) **The effect of the probability of occurrence of high-priority tasks:*** The number of vehicles I is fixed to 150, P_H and RAT take the same value as above. In Fig. 3 (a), RHATP sum utility is the smallest, with the increase of P_H, the difference between RHATP and NTP gradually increases. The main reason is the increase of the number of high-priority tasks, NTP compared to RHATP high-priority task delay will be larger, weighted calculation of the sum utility gap will be larger. At the same time this will gradually reduce the gap between NRH and NTP. Whereas Fig. 3 (b) shows that RHATP can control the high-priority task delay within a small range. High-priority tasks of NTP have high delay because it does not differentiate between task priorities.

*3) **Relationship between sum utility and RAT:*** By setting P_H to 0.3 and I to 150, RAT is changed by varying the number of task-vehicles in each region without changing the total number of vehicles in each region and the total number of task-vehicles on the roads. Since the total number of task-vehicles remains constant, the number of task-vehicles in each RSU region is nearly the same in the later stage as the RAT increases. The number of vehicles that can be matched successfully gradually reaches saturation, which leads to the system's offloading decision and resource allocation strategy not changing much either. Therefore, from Fig. 4, it can be seen that the sum utility of the three scenarios increases with the increase of RAT and gradually tends to be stable. In the early stage, the increase of RAT leads to the increase of the number of vehicles that can be matched, and the idle resources that can be called by the low-load RSU also increases. And the gain in transmission delay and computation delay brought by hopping offloading will increase, so the difference between RHATP and the other two strategies in the early stage will increase.

VI. Conclusion

This paper incorporates MEC techniques to achieve the goal that the computational resources of the RSU can be utilized to assist the vehicle in processing data from various sensors. In response to the communication congestion caused by the uneven distribution of task-vehicles on the road, this paper utilizes the marriage matching model in game theory to establish V2V matching pairs, which call the computational and communication resources of RSUs in the low-load region of the task with the help of relay-vehicles. At the same time, the priority of the tasks is considered and more resources are tilted to the high-priority tasks as a way to ensure the low delay of the high-priority. Convex optimisation is used to allocate resources to find the optimal solution of the optimisation problem. Due to the discrete nature of the channel and the large number of variables, this paper uses CVXPY to invoke the academic version of CPLEX to solve the problem. The experimental results show that the algorithm proposed in this



(a) Sum utility (b) Task average delay

Fig. 3: The effect of the occurrence probability P_H of high-priority tasks

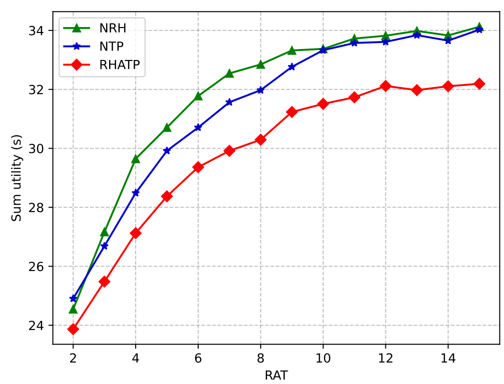


Fig. 4: Relationship between sum utility and *RAT*

paper can reduce the system delay and ensure the sum utility under the condition of guaranteeing the delay of tasks with high-priority.

In the future, we can design the vehicle's motion scene more complex, such as intersections, curves, etc., to improve the adaptability of the model, and we can also partially offload the tasks to further improve the resource utilization.

References

[1] Z. Wei, F. Liu, C. Masouros, N. Su and A. P. Petropulu, "Toward Multi-Functional 6G Wireless Networks: Integrating Sensing, Communication, and Security," in IEEE Communications Magazine," *IEEE Communications Magazine*, vol.60, no.4, pp.65-71, 2022.

[2] Z. Zhu et al., "DRL-based STAR-RIS-Assisted ISAC Secure Communications," *2023 International Conference on Ubiquitous Communication (Ucom), Xi'an, China*, 2023, pp. 127-132, doi: 10.1109/Ucom59132.2023.10257639.

[3] Z. Zhu et al., "Intelligent Reflecting Surface-Assisted Wireless Powered Heterogeneous Networks," *IEEE Transactions on Wireless Communications*, vol. 22, no. 12, pp. 9881-9892, 2023.

[4] Y. Cui, F. Liu, X. Jing and J. Mu, "Integrating Sensing and Communications for Ubiquitous IoT: Applications, Trends, and Challenges," *IEEE Network*, vol.35, no.5, pp.158-167, 2021.

[5] J. Choi, V. Va, N. Gonzalez-Prelcic, R. Daniels, C. R. Bhat and R. W. Heath, "Millimeter-Wave Vehicular Communication to Support Massive Automotive Sensing," *IEEE Communications Magazine*, vol.54, no.12, pp.160-167, 2016.

[6] X. Cheng, D. Duan, S. Gao and L. Yang, "Integrated Sensing and Communications (ISAC) for Vehicular Communication Networks (VCN)," *IEEE Internet of Things Journal*, vol.9, no.23, pp.23441-23451, 2022.

[7] J. Cao, W. Feng, N. Ge and J. Lu, "Delay Characterization of Mobile-Edge Computing for 6G Time-Sensitive Services," *IEEE Internet of Things Journal*, vol.8, no.5, pp.3758-3773, 2021.

[8] J. Santos, T. Wauters, B. Volckaert and F. De Turck, "Towards Low-Latency Service Delivery in a Continuum of Virtual Resources: State-of-the-Art and Research Directions," *IEEE Communications Surveys and Tutorials*, vol.23, no.4, pp.2557-2589, 2021.

[9] T. Zhao, Y. Zhu, Y. Ding *et al.*, "Research on content-aware classification offloading algorithm based on mobile edge computing in vehicular networking," *Journal of Electronics and Information Technology*, vol.42, no.1, pp.20-27, 2020.

[10] D. Cao, K. Zeng, J. Wang, P. K. Sharma, X. Ma, Y. Liu, S. Zhou, "BERT-Based Deep Spatial-Temporal Network for Taxi Demand Prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol.23, no.7, pp.9442-9454, 2022.

[11] D. Cao, R. Dai, J. Wang, B. Ji, X. Ma, Y. Liu, S. Zhou *et al.*, "Fast Visual Tracking with Squeeze and Excitation Region Proposal Network," *HUMAN-CENTRIC COMPUTING AND INFORMATION SCIENCES*, vol.13,

pp.1-20, 2023.

[12] B. Gao, Z. Zhou, F. Liu, F. Xu and B. Li, "An Online Framework for Joint Network Selection and Service Placement in Mobile Edge Computing," *IEEE Transactions on Mobile Computing*, vol.21, no.11, pp.3836-3851, 2022.

[13] Y. Liu, M. Li, A. Liu, J. Lu and T. X. Han, "Information-Theoretic Limits of Integrated Sensing and Communication With Correlated Sensing and Channel States for Vehicular Networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 9, pp. 10161-10166, 2022.

[14] Q. Liu, R. Luo, H. Liang and Q. Liu, "Energy-Efficient Joint Computation Offloading and Resource Allocation Strategy for ISAC-Aided 6G V2X Networks," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 1, pp. 413-423, 2023.

[15] X. Li, L. Ma, R. Shankaran, Y. Xu and M. A. Orgun, "Joint Power Control and Resource Allocation Mode Selection for Safety-Related V2X Communication," *IEEE Transactions on Vehicular Technology*, vol.68, no.8, pp.7970-7986, 2019.

[16] X. He, C. Guo and B. Liao, "Spectrum and Power Allocation for Vehicular Networks with Diverse Latency Requirements," *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Beijing, China*, pp.1-5, 2019.

[17] F. Li, H. Zhang, W. Zi, "MEC-based V2X cooperative caching and resource allocation in Telematics," *Journal of Communications*, vol.42, no.2, pp.26-36, 2021.

[18] Z. Sharif, L. T. Jung, I. Razzak and M. Alazab, "Adaptive and Priority-Based Resource Allocation for Efficient Resources Utilization in Mobile-Edge Computing," *IEEE Internet of Things Journal*, vol.10, no.4, pp.3079-3093, 2023.

[19] R. Wang, X. Jiang, Y. Zhou, Z. Li, D. Wu, T. Tang, "Multi-agent reinforcement learning for edge information sharing in vehicular networks," *Digital Communications and Networks*, vol.8, no.3, pp.267-277, 2022.

[20] Zhao J , Sun X , Li Q *et al.*, "Edge Caching and Computation Management for Real-Time Internet of Vehicles: An Online and Distributed Approach," *IEEE Transactions on Intelligent Transportation Systems*, dio: 10.1109/TITS.2020.3012966, 2020.

[21] L. Liu, C. Chen, J. Feng *et al.*, "Joint intelligent optimisation of task offloading and service caching in in-vehicle edge computing," *Journal of Communications*, vol.42, no.1, pp.18-26, 2021.

[22] D. Cao, Y. Zhang, Z. Dian *et al.*, "V2X multi-node cooperative distributed offloading strategy," *Journal of Communications*, vol.43, no.2, pp.185-195, 2022.

[23] Y. Ren, F. Zhu, J. Wang, P. K. Sharma and U. Ghosh, "Novel Vote Scheme for Decision-Making Feedback Based on Blockchain in Internet of Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol.23, no.2, pp. 1639-1648, 2022.

[24] C. Yang, W. Lou, Y. Liu, S. Xie *et al.*, "Resource Allocation for Edge Computing-Based Vehicle Platoon on Freeway: A Contract-Optimization Approach," *IEEE Transactions on Vehicular Technology*, dio:10.1109/TVT.2020.3039851, 2020.

[25] H. Zhang, Y. Cheng, K. Liu *et al.*, "Mobility Management Strategies for Integrating Mobile Edge Computing and Content Delivery Networks in Telematics," *Journal of Electronics and Information Technology*, vol.42, no.6, pp.1444-1451, 2020.

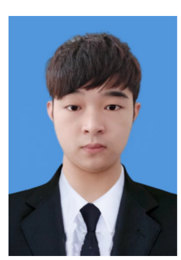
[26] Z. Zhang, X. Li, D. Liu, T. Luo and Y. Zhang, "Trajectory Data Driven V2V/V2I Mode Switching and Bandwidth Allocation for Vehicle Networks," *IEEE Wireless Communications Letters*, vol.9, no.6, pp.795-798, 2020.

[27] T. Zhou, Y. Yue, D. Qin, X. Nie, X. Li and C. Li, "Joint Device Association, Resource Allocation, and Computation Offloading in Ultradense Multidevice and Multitask IoT Networks," *IEEE Internet of Things Journal*, vol.9, no.19, pp.18695-18709, 2022.

[28] X. Lai, X. Ma,Y. Bai *et al.*, "Dynamic reactive power optimisation based on mixed integer second order cone programming," *Automation of Electric Power Systems*, vol.41, no.17, pp.37-42, 2017.

Dun Cao (Member, IEEE) received the B.S. and M.S. degree from Central South and Hunan University in 2001, 2006 respectively. She received the Ph.D. degree from Changsha University of Science and Technology in 2017. She is an associated professor in School of Computer and Communication Engineering, Changsha University of Science and Technology. And she is the member of Xiangjiang Laboratory, Changsha 410205, China. She has published more than 30 journal and conference papers. And her research interests include vehicular networks and MIMO wireless communications etc.

Meihua Wu is currently pursuing the M.S. degree in the Changsha University of Science and Technology. Her current research interests is mobile edge computing for Internet of vehicles.



Ning Gu is currently pursuing the M.S. degree in the Changsha University of Science and Technology. His current research interests is mobile edge computing for Internet of vehicles.

R. Simon Sherratt (M97-SM02) was born in Heswall, UK. He received his B.Eng. degree in Electronic Systems and Control Engineering from Sheffield City Polytechnic UK in 1992, M.Sc. degree in data telecommunications and Ph.D. degree in video signal processing from the University of Salford in 1994 and 1996. Since 1996, he has been a Lecturer in Electronic Engineering at the University of Reading where he is now a senior lecturer in consumer electronics and currently head of Electronic Engineering. His research topic is signal processing in consumer electronic devices concentrating on equalization, communications layer 1, DSP architectures and adaptive signal processing.

Eur Ing Dr. Sherratt is a senior member of the IEEE, member of the IEEE Consumer Electronics Society AdCom (2003-2005, 2006-2008) holding the International Symposium on Consumer Electronics Liaison officer post (2005-) and the Society awards chair post (2006-), member of the IEEE Transactions on Consumer Electronics publications committee (2004-), IEEE International Conference on Consumer Electronics ExCom and vice-technical chair (2006), chair of the IEEE International Symposium on Consumer Electronics 2004 and committee member for 2002, 2003, 2005 and 2006, founder and current chair of the IEEE UKRI Consumer Electronics and Broadcast Technology Joint Chapter. He received the IEEE Chester Sall 1st place best Transactions paper award in 2004 and the best paper in the IEEE International Symposium on Consumer Electronics 2006.

Uttam Ghosh (Senior Member, IEEE) received the Ph.D. degree in electronics and electrical engineering from IIT Kharagpur, India, in 2013. He was an Assistant Professor at the Department of Computer Science, Vanderbilt University. He was a Post-Doctoral Researcher at the University of Illinois in Urbana–Champaign, Fordham University, and Tennessee State University. He is an Associate Professor of cybersecurity with the Department of Computer Science and Data Science, Meharry Medical College (MMC), Nashville, TN, USA. His main research interests include cybersecurity, computer networks, wireless networks, machine learning, health informatics, and software-defined networking. He is a member of Sigma Xi and ACM.(Based on document published on 16 December 2022).

Pradip Kumar Sharma (Senior Member, IEEE) received the Ph.D. degree in CSE from the Seoul National University of Science and Technology, Seoul, South Korea, in 2019. He worked as a Post-Doctoral Research Fellow with the Department of Multimedia Engineering, Dongguk University, Seoul. He is currently an Assistant Professor in cybersecurity with the Department of Computing Science, University of Aberdeen, U.K. His current research interests are focused on the areas of cybersecurity, blockchain, edge computing, SDN, and the IoT security. He has published many technical research articles in leading journals from IEEE, Elsevier, Springer, and MDPI. Some of his research findings are published in the most cited journals. He has been an expert reviewer for IEEE Transactions, Elsevier, Springer, and MDPI journals and magazines. He has also been invited to serve as a Technical Program Committee Member and the Chair for several reputed international conferences, such as IEEE CNCC 202, CSA 20202, IEEE ICC2019, IEEE MENACOMM'19, and 3ICT 2019. He is currently an Associate Editor of Human-centric Computing and Information Sciences (HCIS), Electronics (MDPI), and Journal of Information Processing Systems (JIPS) journals. He has been serving as a Guest Editor for international journals of certain publishers, such as IEEE, Elsevier, Springer, MDPI, and JIPS. He is listed in the world's Top 2% Scientists for citation impact during the calendar year 2019 by Stanford University. Also, he received a top 1% reviewer in computer science by Publons Peer Review Awards 2018 and 2019, Clarivate Analytics.