Received 16 June 2022; revised 17 July 2022; accepted 26 July 2022. Date of publication 5 August 2022; date of current version 18 August 2022.

Digital Object Identifier 10.1109/OJCOMS.2022.3195703

Energy-Efficiency Computation Offloading Strategy in UAV Aided V2X Network With Integrated Sensing and Communication

QIAN LIU [1,2,3], HAIRONG LIANG [1,2], RUI LUO [1,2], AND QILIE LIU [1,2]

[1]School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

[2]Chongqing Key Laboratory of Mobile Communication Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

[3]College of Electronical and Information Engineering, Nanchang Institute of Technology, Nanchang 330000, China

CORRESPONDING AUTHOR: Q. LIU (e-mail: liuql@cqupt.edu.cn)

This work was supported in part by the Basic and Advanced Research Projects of CSTC under Grant cstc2019jcyj-zdxmX0008; in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K201900605; and in part by the Doctoral Initial Funding of Chongqing University of Posts and Telecommunications under Grant A2021-195(E012A2021195).

ABSTRACT Integrated Sensing and Communications (ISAC) technology can jointly design radio sensing and communication functionalities, which enable 6G Ere to have the ability to "see" the physical world rather than communication-only. Benefitting from ISAC, vehicular-to-everything (V2X) networks may efficiently complete high-precision traffic environment perception. Furthermore, with the assistance of flexibly deployed Unmanned Aerial Vehicles (UAVs), the V2X networks overcome the limited sensing range of sensors equipped on a vehicle and guarantee safe driving. This paper proposes an energy-efficient computation offloading strategy for multiple sensor data fusion in UAV Aided V2X Network supported by Integrated Sensing and Communication. Firstly, a vehicle-UAV cooperative perception architecture is proposed to perceive a wide range of traffic environments. Secondly, we introduce a computation offloading strategy jointly considering offloading decisions and dynamic computing resource allocation. Finally, a successive convex approximation (SCA) algorithm transforms a non-convex formulation problem into a tractable convex approximation problem. The simulation results show that the strategy proposed in this paper reduces the UAV energy consumption and data fusion task processing delay.

INDEX TERMS Unmanned aerial vehicles, vehicular-to-everything, mobile computing, integrated sensing and communication, successive convex approximation.

I. INTRODUCTION

THE SENSING and communication functionalities are possibly jointly designed to utilize wireless/hardware resources efficiently and to assist each other for mutual benefits. Therefore, Integrated Sensing and Communications (ISAC) technique is widely recognized for triggering a paradigm shift in the digital world [1], [2]. Benefitting from the development of the ISAC technique, a gradual integration of the physical and digital worlds is perceived to become a reality in the 6G era. As a result, a large number of real-time sensing applications have emerged, such as intelligent vehicles [3], disaster monitoring [4], and precision agriculture [5]. Among them, ISAC for the 6G vehicular-to-everything (V2X) network has recently attracted a lot of attention in the automotive industry and academia.

In the V2X network with ISAC, intelligent vehicles comprise many components such as On-Board Units (OBUs) and multiple sensors. OBUs are placed on vehicles to provide high-quality communication services with V2X networks by connecting roadside unit (RSU). These sensors (e.g., laser radars, millimeter-wave radars, and cameras) can accurately perceive the surrounding environment to respond to objects and vehicles on the road and adjust to emergencies. However, many accidents involve visual occlusions of cars or vulnerable road users. The instrumenting of individual vehicles with sensors in current intelligent driving and advanced driver

assistance system cannot fully solve similar occlusions [6]. Therefore, sensor data fusion among multi-vehicles is crucial for the moving safety of intelligent vehicles.

In the current research, there are mainly two solutions. One is the roadside perception technology to assist vehicles, and the other is the collaborative perception data sharing within a platoon of vehicles. However, the first solution can only be applied to urban streets and intersections equipped with roadside sensors. The second solution has higher requirements for synchronizing the driving trajectory and is only suitable for some users. In order to overcome the difficulty of applicability of the above solutions, we propose a vehicle-UAV cooperative perception architecture. It can take advantage of the high mobility and extensive service coverage of Unmanned Aerial Vehicles (UAVs) to provide accompanying perception fusion support for multiple vehicles.

Nevertheless, limited by the computing resources and energy of UAVs, the multi-source information fusion computing tasks cannot be adequately performed at UVA. Therefore, edge computing technology [7], [8], [9] will be used to perform reasonable computing task offloading for these computing tasks, thereby overcoming the above shortcomings and providing an additional benefit of reducing task processing latency. There are many works on UAV-assisted MEC systems. In [10], the authors investigate a UAV-assisted wirelessly powered MEC system that minimizes the total energy consumption of the UAV by optimizing the CPU frequency and the UAV trajectory. In [11], the authors investigate the energy efficiency of a UAV-assisted MEC system aimed at minimizing the hovering energy and computational energy consumption of the UAV. In [12], the authors studied the joint UAV trajectory and computed the offloading optimization problem to minimize the time delay of the offloading task. In [13], the authors study a UAV-assisted edge computing system whose goal is to minimize the sum of the maximum latency, task offload rate, and UAV flight trajectory for all users in each time slot by jointly optimizing user scheduling. While these studies consider the resource scheduling problem under user latency requirements and energy consumption under UAV constraints, they do not consider the need to offload computational tasks for multi-sensing data fusion in V2X-UAV networks with integrated sensing and communication.

Some papers focus on multi-sensory data fusion in cooperative perception traffic scenarios. Sun *et al.* [14] propose a novel MEC model that considers the MEC server as having input and output data for all computing tasks and communicating with all users through a shared wireless link. Zhang *et al.* [15] present the joint problem of data offloading decisions, fusion scope, and computational resource allocation to maximize the system fusion gain. A computational offloading and resource allocation optimization method for AVR with hybrid sensing data fusion algorithm (CoroAVR) is proposed in [16]. It processes real-time data in fog-edge computing to maximize system throughput and spectrum utilization. It is worth noting that the above work focuses on minimizing latency or maximizing throughput, ignoring the need for energy efficiency and perceived data variability. And in the face of wireless network scenarios where infrastructure is limited or unavailable, such as disaster area relief, traditional terrestrial edge computing is no longer satisfactory for users' needs.

Unlike the above studies, we focus on the proposed computational offloading problem involving sensor data fusion in UAV-aided V2X networks. A single vehicle has a limited sensing range and cannot perceive the complete environmental information. Therefore, multi-vehicle sensing data fusion on UAVs provides real-time and accurate map scene information for vehicle travel. Since UAV computational resources are limited, we address the lack of computational resources by offloading some of the UAV data to RSU for processing. On the one hand, the urgency and importance of the sensor data (e.g., latency, accuracy) are considered to have specific differences. On the other hand, the computational resources of the system are considered to be mainly concentrated on the ground RSU side. Therefore, we propose an energy-efficient computational offloading strategy in this paper. It can prioritize the sensed data and perform a weighted allocation of computational resources on the ground RSU side according to the vehicle's perception data priority. Considering the limited energy consumption of UAVs and the timeliness of sensed data, we minimize the energy consumption and total user latency weighted sum of UAVs by jointly optimizing the bandwidth resources, computational resources and offloading decisions of the system. The key contributions of this paper are briefly summarized as follows:

1) We propose a cooperative vehicle-UAV perception architecture. UAVs equipped with sensors assist vehicles in perceiving a wide range of traffic environments. And to cope with the rapidly growing computational volume of data fusion tasks and the latency-sensitive nature of vehicular network services, MEC techniques are integrated into the architecture to compensate for the lack of computational resources. The proposed strategy is designed based on this architecture.
2) Considering the differential requirements of UAV energy consumption and perceived fused data, this paper models the offloading decision and resource allocation problem for data fusion computing tasks as a task priority-based latency and energy consumption weighting and minimization problem to achieve low latency response and energy efficient.
3) We propose an SCA convex approximation algorithm to transform the formulation non-convex problem into a tractable convex approximation problem. Then, the optimal solution is obtained through successive iterations.

The rest of this paper is organized as follows. In Sections II, the system model is introduced. In Section III, we mathematically formulate the optimization problem. In

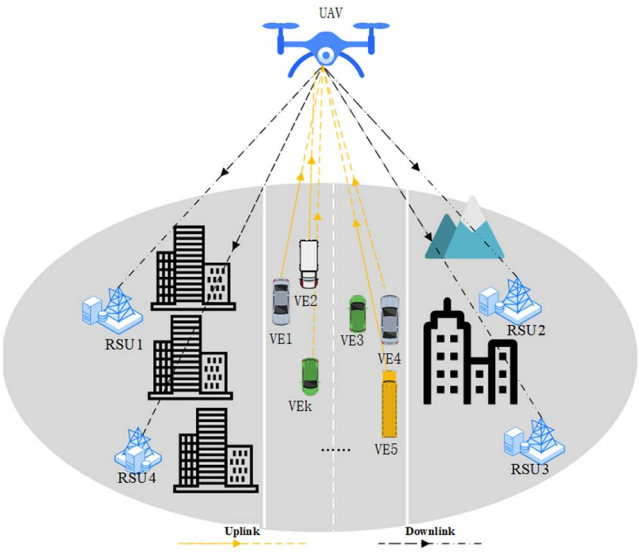

FIGURE 1. System Model.

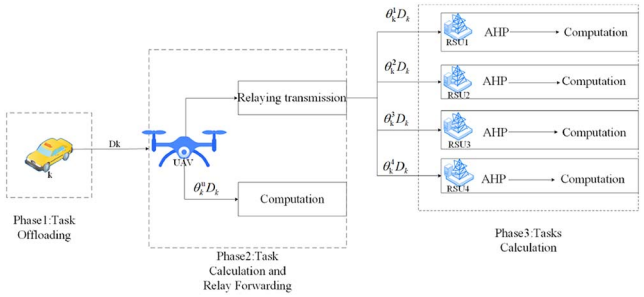

FIGURE 2. Three phases for computation task execution.

Section IV, we introduce the proposed method of solving the optimization problem. Section V presents the numerical results and discusses the performance of the proposed scheme. Finally, we conclude the paper in Section VI.

II. SYSTEM MODEL

The model of UAV-assisted edge computing system supported by ISAC is shown in Fig. 1. In this system, UAVs loaded with MEC servers and sensors and ground RSUs assist in providing communication and multi-sensory data fusion services to K smart vehicles on a roadway, where k $\in$ K, $k = \{1, 2, \ldots, K\}$. The vehicle periodically collects sensing data and generates computational tasks for data fusion $\xi_k = \langle S_k, D_k, C_k \rangle$, where D_k refers to the data size of the computational tasks to be processed by the vehicle k, C_k is the number of cycles that the central processing unit (CPU) needs to run for the vehicle k to process 1 bit of task data, and S_k indicates the service level of the vehicle computational tasks, with higher levels indicating their higher importance. Without loss of generality, we use a three-dimensional Cartesian coordinate system to represent the positions of the vehicle, RSUs and UAV. We assume that the position of the vehicle k is known and stationary with respect to the UAV, where the coordinates of the vehicle k are denoted by $q^k = (x^k, y^k, 0)$. The coordinates of the RSUi are denoted by $q^i = (x^i, y^i, 0)$, which are also fixed and known. We assume that the UAV flies at a fixed altitude H, effectively avoiding all kinds of collisions, and its position coordinates are $q^u = (x^u, y^u, H)$.

In the architecture of the system, the smart vehicle keeps in communication with the UAV through a wireless link and sends the various sensor data it collects to the UAV. Then, both sides of the UAV perform data fusion of all the collected sensory data to form a traffic perception environment map in its coverage area and send it to the service vehicle. On the vehicle side, the vehicle-based itself performs a secondary fusion of the wide-area traffic perception environment map and its motion-aware data to support finer and more accurate driving decisions and improve the safety of the vehicle.

The traffic environment perception map data processing is divided into three mainly phases: perception task data upload, task data processing on the UAV side and task data processing on the RSU side, as shown in Fig. 2. Specifically, in the first stage, the vehicle k first uploads all the task data to be processed to the UAV. The first phase ends when the UAV receives all the task data from the vehicle k.

Similar to the literature [17], we assume that there is complete granularity in the computational task data partition and the vehicle computational task data can be divided into arbitrary proportions for offloading. Therefore, in the second stage we use a partial offloading model where the UAV is able to divide the computational tasks of vehicle k into five parts. the UAV processes a part locally and forwards the remaining part to the ground RSUs for processing via relay.

Finally, in the third stage, the ground base station performs priority weight calculation for some of the tasks received from vehicle k and allocates computational resources for processing according to the priority weight. In most computationally intensive applications, such as face recognition and video analysis, the delay and energy consumption required to process the results of the computational tasks back to the vehicle side is much smaller than the delay and energy consumption of the offloading of the vehicle computational tasks, so we ignore them [18].

A. COMMUNICATION MODEL

When the UAV flies at a certain altitude to communicate with the vehicle, the communication channel between the UAV and the vehicle k and the RSUs is dominated by the line of sight transmission (LoS) link [19], so the channel between the UAV and the vehicle and the channel between the UAV and the RSUs uses a free-space path loss model. The channel gain between the vehicle and the UAV is expressed as

$$h_k^u \triangleq \alpha_0 \left(d_k^u\right)^{-2} = \frac{\alpha_0}{H^2 + \left\| q^u - q^k \right\|^2}, \tag{1}$$

where α_0 denotes the received power when the reference distance is 1 m and the transmission power is 1 W, d_k^u denotes the transmission distance between the mobile user

and the UAV, and $\|x\|$ denotes he Euclidean parametrization of the vector. Similarly, the link gain from the UAV to the ground RSUi can be expressed as

$$h_i^u \triangleq \alpha_0 \left(d_i^u\right)^{-2} = \frac{\alpha_0}{H^2 + \left\|q^u - q^i\right\|^2}, \tag{2}$$

where d_i^u denotes the downlink transmission distance from the UAV to RSUi.

To eliminate the signal interference between the vehicles and to simplify the system model, the frequency division multiple access (OFDMA) technique is used for data transmission between the UAV and the vehicle. The data transmission rate between the UAV and the vehicle equipment depends on the physical channel characteristics and the corresponding channel gain. We use r_u^k to denote the data transmission rate between the UAV and the vehicle, which, according to Shannon's second theorem, can be expressed as

$$r_k^u = B_k^u log_2\left(1 + \frac{h_k^u P_k}{\sigma^2}\right), \tag{3}$$

where B_u^k denotes the bandwidth resources allocated by the UAV to each vehicle, and P_k and σ^2 denote the transmit power of the vehicle and the noise power at the UAV, respectively. We use B^{UAV} to denote the bandwidth resources that can be allocated by the UAV, so B_k^u is limited by B^{UAV} and the number of vehicles K

$$\sum_{k=1}^{K} B_k^u \leq B^{UAV}. \tag{4}$$

Consistent with what was mentioned above, all vehicles offload computational tasks to the UAV. The transmission delay of vehicle k transmitting data to the UAV is related to the uplink transmission rate and can be expressed as

$$t_k^u = \frac{D_k}{r_k^u}. \tag{5}$$

For simplicity, we consider that the noise power at the UAV and at the RSU is the same [20]. However, it is also very easy to scale when the noise power spectral density between the two is different. Therefore, the downlink transmission rate between UAV and RSUi can be calculated as

$$r_i^u = B_i^u \log_2\left(1 + \frac{h_i^u P^{uT}}{\sigma^2}\right), \tag{6}$$

where B_i^u denotes the transmission bandwidth between the RSU and the UAV, P^{uT} is the transmit power of the UAV.

After UAV finishes receiving all the data transmitted by vehicles, it determines the partial offloading ratio of vehicles' computational tasks according to the existing resource status of the system. $\theta_k^u \in [0, 1]$ and $\theta_k^i \in [0, 1]$ denote the ratio of vehicles' computational tasks processed by servers at UAV side and RSU side, respectively, and $\theta_k^u + \sum_{i \in I} \theta_k^i = 1$. The UAV can consider further offloading the computational tasks to the RSU with more computational power for processing in order to achieve a lower latency. The transmission latency between the UAV and the RSU can be calculated as

$$t_i^{uk} = \frac{\theta_k^i D_k}{r_i^u}. \tag{7}$$

B. COMPUTATION MODEL

We assume that processing is performed on the server of the UAV or RSU only when the task data transfer is completed. Moreover, by performing dynamic voltage and frequency scaling (DVFS), the server is able to dynamically allocate its computational resources based on the type or number of tasks arriving [21]. Let f_k^u denote the CPU resources allocated by the UAV server to the vehicle k computational tasks and F^{UAV} denote the total CPU computational resources of the UAV. Thus we can obtain the computational latency and computational resource constraints for the tasks on the UAV side

$$t_{cp}^{uk} = \frac{\theta_k^u D_k C_k}{f_k^u}, \tag{8}$$

$$\sum_{k=1}^{K} f_k^u \leq F^{UAV}. \tag{9}$$

When the data from the UAV offload is received, the RSU starts the computational processing process. Compared to UAV, RSU have larger computational resources. Therefore, the allocation of RSU computational resources will dominate the completion delay of vehicle-aware computational tasks. This also means that the proper allocation of computational resources to RSU will directly affect the judgment decision of vehicle perception. Thus we use hierarchical analysis (AHP) [22] to analyze the priority weight of each vehicle's data fusion task among all data fusion tasks at the current phase. Then, the computational resources of the RSU are allocated according to the priority weights so as to meet the differentiated demands of different vehicles' computational tasks. First, we construct the hierarchical model with the priority weight of the vehicle, the calculation task parameters S_k, D_k, C_k and the vehicle k as the target layer, the criterion layer and the solution layer, respectively. Then, according to the hierarchical model, the judgment matrix $A = (a_{ij})_{n*n}$ for each layer other than the target layer is constructed, where

$$a_{ij} = \begin{cases} \frac{1}{a_{ij}} = rho, rho = \{1, 2, \ldots, 9\} \\ 1, i = j. \end{cases} \tag{10}$$

Solve the relative weight of the judgment matrix to a certain element in the upper level by the characteristic root method $\omega = (\omega_1, \omega_2, \ldots, \omega_n)^T$

$$A\omega = \lambda_{max}\omega. \tag{11}$$

The relative weight vector of the criterion layer with respect to the target layer is expressed as

$$\Lambda = [\omega_S, \omega_D, \omega_C]^T. \tag{12}$$

Under the judgment criteria S_k, D_k, C_k, the computed task-shaped weight matrix for each vehicle k can be expressed as

$$\Delta = \begin{bmatrix} \omega_1^S & \omega_1^D & \omega_1^C \\ \omega_2^S & \omega_2^D & \omega_2^C \\ \vdots & \vdots & \vdots \\ \omega_k^S & \omega_k^D & \omega_k^C \end{bmatrix}. \tag{13}$$

After the consistency check, the corresponding user computes the task priority weight vector L can be calculated as

$$\mathrm{L} = \Delta * \Lambda = [l_1, l_2, \ldots, l_k]^T. \tag{14}$$

We let f_k^i denote the CPU resources allocated by the RSUi server to vehicle k computation tasks, and let F^i denote the total CPU computation resources of RSUi. Thus we can obtain the computational latency and computational resource constraints for the tasks on the RSU side

$$t_{cp}^{ik} = \frac{\theta_k^i D_k C_k}{f_k^i}, \tag{15}$$

$$\sum_{k=1}^{K} f_k^i \le F^i, \tag{16}$$

$$f_k^i \le l_k F^i. \tag{17}$$

C. ENERGY CONSUMPTION MODEL

Due to the limited energy storage of UAVs, the energy consumption management of UAVs is crucial in order to extend the operating hours of UAV-assisted edge computing systems. Therefore, in this paper, we focus on the energy consumption of UAV. During the whole computational task completion process, UAV energy consumption consists of computational task reception energy consumption, computational task execution energy consumption, and computational task transmission energy consumption. The hovering energy consumption of the UAV is not considered in our model because it is related to the performance of the UAV itself and is not relevant to our offloading decision [23].

1) *UAV computational energy consumption:* We use β to denote the effective capacitance factor of the UAV, which depends on the chip structure of the CPU of the UAV [24]. Similar to the literature [25], where the computational energy consumption f_k^u is proportional to the third power of the UAV, the UAV computational energy consumption is expressed as the product of the energy consumption level of the UAV processing the offloading task and the time.

$$E_k^{cp} = \beta \left(f_k^u\right)^3 t_{cp}^{uk} = \beta \left(f_k^u\right)^2 \theta_k^u D_k C_k. \tag{18}$$

2) *UAV reception energy consumption:* The energy consumption of the UAV to fully receive the data of the computing tasks transmitted by the vehicle via the uplink is

$$E_k^R = P^{uR} t_k^u = \frac{D_k P^{uR}}{r_k^u}, \tag{19}$$

where P^{uR} is the received power of the UAV.

3) *UAV transmission energy consumption:* The energy consumption for transferring part of the offload data to the RSU via UAV is

$$E_{ki}^T = P^{uT} t_i^{uk} = \frac{\theta_k^i D_k P^{uT}}{r_i^{uk}}. \tag{20}$$

III. OPTIMIZATION PROBLEM FORMULATION

In the system presented in this paper, the vehicle computing task is offloaded to the UAV and RSUs for parallel processing. The computing task completion time of vehicle k is set to the latest calculation and processing time of UAV and RSUs terminals. The total delay of computing task processing of vehicle k can be expressed as

$$\mathrm{T}_k = t_k^u + \max\left\{t_{cp}^{uk}, t_i^{uk} + t_{cp}^{ik}\right\}. \tag{21}$$

In addition, the energy consumption of the UAV during the whole process of handling the vehicle k calculation task is expressed as

$$\mathrm{E}_k = \left(E_k^R + E_k^{cp} + \sum_{i \in I} E_{ki}^T\right). \tag{22}$$

Based on the above discussion, we aim to minimize the weighted sum of the total time delay and UAV energy consumption of ISAC-supported UAV-assisted edge computing systems serving vehicle computing task processing by jointly optimizing system bandwidth resources, and offloading decisions, and computational resources of UAVs and RSUs. Corresponding optimization problems are expressed as follows

$$P1: \min_{B_k^u, \theta, f_k^u, f_k^i} \sum_{k \in K} \delta_1 E_k + \delta_2 T_k$$

$$s.t. \sum_{\mathrm{k} \in K} B_k^u \le B^{UAV}, \tag{C1}$$

$$\sum_{\mathrm{k} \in K} \mathrm{f}_k^u \le F^{UAV}, \tag{C2}$$

$$\sum_{\mathrm{k} \in K} \mathrm{f}_k^i \le F^i, \forall i, \tag{C3}$$

$$\theta_{\mathrm{k}}^u + \sum_{i \in I} \theta_k^i = 1, \forall \mathrm{k}, \tag{C4}$$

$$0 \le \theta_{\mathrm{k}}^i \le 1, \forall \mathrm{k}, \mathrm{i}, \tag{C5}$$

$$0 \le \theta_{\mathrm{k}}^u \le 1, \forall \mathrm{k}, \tag{C6}$$

$$0 \le \mathrm{f}_k^i \le l_k F^i, \forall \mathrm{k}, \mathrm{i}, \tag{C7}$$

$$\mathrm{B}_k^u \ge 0, \forall \mathrm{k}, \tag{C8}$$

$$\mathrm{f}_k^u \ge 0, \forall \mathrm{k}, \tag{C9}$$

where δ_1 and δ_2 are weighted sum parameters between UAV energy consumption and total latency. C1, C2, C3, C8, and C9 ensure that the allocated communication bandwidth resources, UAV and RSU computational resources are non-negative and do not exceed their upper limits. C7 indicates that the RSU computational resources allocated to individual

vehicles are also limited by the product of priority weights and total RSU computational resources. C4, C5, and C6 indicate that the fusion data task offload is handled entirely by the UAV and RSU, and the partial offload ratio is between 0 and 1.

IV. SCA-BASED OPTIMIZATION ALGORITHM

Problem P1 is difficult to solve directly given the nonconvexity of the objective function and the large number of constraint limitations that exist for this problem. To address this problem, this paper approximates the nonconvex terms in the problem to convex terms by using a successive convex approximation method using a given number of local iteration points. When the problem is transformed into a convex problem, the suboptimal solution of the problem is obtained by the iterative algorithm proposed in this paper.

According to [26], in case the objective function has a product structure such as $\mathrm{U}(x) = h_1(x)h_2(x)$, and $h_1(x)$, $h_2(x)$ are nonnegative and convex, the convex approximation function of $\forall y \in X$, $U(x)$ can be expressed as

$$\widetilde{U}(x,y) = h_1(x)h_2(x) + h_1(y)h_2(y) + \frac{\tau}{2}(x-y)^T A(y)(x-y), \tag{23}$$

where τ is a positive constant, $\mathrm{A}(y)$ is a positive definite matrix. We observe for problem P1 that the nonconvex term E_k^{cp} of its objective function can be rewritten as the product of two nonnegative and convex functions. By making $\mathrm{h}_1(\theta_k^u) = \theta_k^u, h_2(f_k^u) = (f_k^u)^2$, E_k^{cp} can be rewritten as

$$E_k^{cp} = \beta D_k C_k h_1(\theta_u) h_2\left(f_k^u\right). \tag{24}$$

Apparently $\mathrm{h}_1(\theta_u), h_2(f_k^u)$ are non-negative convex functions, and by giving the feasible solutions $((\theta_k^u)_{\mathrm{n}}, (f_k^u)_n)$ for the nth iteration and using formula (23), we can obtain the convex approximation $\widetilde{\mathrm{E}}_k^{cp}$ for E_k^{cp}

$$\widetilde{\mathrm{E}}_k^{cp}\left(\theta_k^u, f_k^u; (\theta_k^u)_n, (f_k^u)_n\right) = \beta D_k C_k \Bigg(\theta_k^u (f_k^u)_n^2 + (\theta_k^u)_n (f_k^u)^2 + \frac{\tau_{\theta_k^u}}{2}\left(\theta_k^u - (\theta_k^u)_n^2\right) + \frac{\tau_{f_k^u}}{2}\left(f_k^u - (f_k^u)_n^2\right)\Bigg). \tag{25}$$

Since there is a nonlinear term $\mathrm{T}_k = t_k^u + \max\{t_{cp}^{uk}, t_i^{uk} + t_{cp}^{ik}\}$ in the objective function, it makes the objective function very complex and difficult to handle. Therefore, this paper introduces the relaxation variable φ_{k} to transform this nonlinear term into a linear term, where $\mathrm{T}_k = t_k^u + \varphi_k$. To make the solution of the original problem identical, φ_{k} should satisfy the following constraints

$$\varphi_{\mathrm{k}} \geq t_{cp}^{uk}, \tag{26}$$

$$\varphi_{\mathrm{k}} \geq \mathrm{t}_i^{uk} + t_{ki}^{cp}. \tag{27}$$

Obviously, the constraints (26), (27) are non-convex constraints. The t_{cp}^{uk} at the right end of the constraint (26) can be reformulated as

$$t_{cp}^{uk} = D_k C_k \left(\frac{1}{2}\left(\frac{1}{f_k^u} + \theta_k^u\right)^2 - \frac{1}{2}\left(\left(\frac{1}{f_k^u}\right)^2 + \left(\theta_k^u\right)^2\right)\right), \tag{28}$$

where $\frac{1}{2}(\frac{1}{f_k^u} + \theta_k^u)^2$ is convex and $-\frac{1}{2}((\frac{1}{f_k^u})^2 + (\theta_k^u)^2)$ is concave, and D_k and C_k are constants that do not affect the concavity and convexity [27]. Given the local point $((\theta_{\mathrm{k}}^u)_n, (f_k^u)_n)$ at the nth iteration, the concave part of the constraint (26) can be expressed as it's upper bound by a Taylor expansion, and the convex approximation $\tilde{t}_{cp}^{uk}$ of t_{cp}^{uk} can be obtained as follows

$$\begin{aligned}\mathrm{t}_{cp}^{uk}(\theta_{\mathrm{k}}^u, f_k^u) &\leq \tilde{t}_{cp}^{uk}(\theta_{\mathrm{k}}^u, f_k^u; (\theta_{\mathrm{k}}^u)_n, (f_k^u)_n) \\ &= D_k C_k \left(\frac{1}{2}\left(\theta_k^u + \frac{1}{f_k^u}\right)^2 - (\theta_k^u)_n^2 - \frac{1}{(f_k^u)_n^2} - (\theta_k^u)_n(\theta_k^u - (\theta_k^u)_n) + \frac{1}{(f_k^u)_n^3}\left(\frac{1}{f_k^u} - \frac{1}{(f_k^u)_n}\right)\right).\end{aligned} \tag{29}$$

Since $\tilde{t}_{cp}^{uk}$ is a convex function with respect to θ_k^u and f_k^u, the constraint (26) is transformed into a convex constraint. Similarly, for t_{cp}^{ik} in constraint (27), we can reformulate it as

$$t_{cp}^{ik} = D_k C_k \left(\frac{1}{2}\left(\frac{1}{f_k^i} + \theta_k^i\right)^2 - \frac{1}{2}\left(\left(\frac{1}{f_k^i}\right)^2 + \left(\theta_k^i\right)^2\right)\right), \tag{30}$$

where $\frac{1}{2}(\frac{1}{f_k^i} + \theta_k^i)^2$ is convex, $-\frac{1}{2}((\frac{1}{f_k^i})^2 + (\theta_k^i)^2)$ is concave, and D_k and C_k are constants that do not affect the concavity and convexity. Given the local point $((\theta_{\mathrm{k}}^i)_n, (f_k^i)_n)$ of the nth iteration, the concave part of the constraint (27) can be expressed as its upper bound by a Taylor expansion, and the convex approximation $\tilde{t}_{cp}^{ik}$ of t_{cp}^{ik} can be obtained as

$$\begin{aligned}\mathrm{t}_{cp}^{ik}\left(\theta_i^k, f_i^k\right) &\leq \tilde{t}_{cp}^{ik}\left(\theta_i^k, f_i^k; \left(\theta_i^k\right)_n, \left(f_i^k\right)_n\right) \\ &= D_k C_k \left(\frac{1}{2}\left(\theta_k^i + \frac{1}{f_k^i}\right)^2 - \left(\theta_k^i\right)_n^2 - \frac{1}{\left(f_k^i\right)_n^2} - \left(\theta_k^i\right)_n\left(\theta_k^i - \left(\theta_k^i\right)_n\right) + \frac{1}{\left(f_k^i\right)_n^3}\left(\frac{1}{f_k^i} - \frac{1}{\left(f_k^i\right)_n}\right)\right).\end{aligned} \tag{31}$$

Since t_{uk}^i is a convex function off θ_k^i and $\tilde{t}_{cp}^{ik}$ is a convex function on θ_k^i and f_k^i, the constraint (27) is converted to a convex constraint.

By the above transformation, the convex approximation problem for problem P1 can be formulated as

$$\begin{aligned} P2: \min_{B_k^u, \theta, f_k^u, f_k^i, \varphi_k} \quad & \sum_{k\in K}\left(\delta_1\left(\tilde{E}_k^{cp} + E_k^R + E_k^T\right) + \delta_2\left(t_k^u + \varphi_k\right)\right) \\ s.t. \quad & C1 \sim C9, \\ & \varphi_{\mathrm{k}} \geq t_i^{uk} + \tilde{t}_{cp}^{ik}, \quad \text{(C10)} \\ & \varphi_{\mathrm{k}} \geq \tilde{t}_{cp}^{uk}. \quad \text{(C11)} \end{aligned}$$

By the given initial iteration point, P2 is a convex optimization problem, so it can be solved by using the CVX

Algorithm 1 The SCA-Based Optimization Algorithm

Input: $x(0)$=[$\theta(0)$;$f_k^i(0)$;$f_k^{uav}(0)$], ε, $\alpha = 0.5$
Output: B_k^u, θ, f_k^u, f_k^i, φ_k;
1: initialize $n = 0$
2: **while** $x(n)$ is not a stationary solution **do**
3: calculate the optimal solution of problem P2
4: update $x(n+1) = x(n) + \mu(\widetilde{x}(n) - x(n))$
5: update $\mu(n) = \mu(n-1)(1 - \alpha\mu(n))$
6: $n = n + 1$
7: **end while**

TABLE 1. System parameters in the simulation.

Parameter	Value	Parameter	Value
K	10	P^{uT}	1 W
H	100 m	D_k	[1,5]Mbit
α_0	-50 dB	C_k	1000cycles/bit
σ^2	-100 dBm	S_k	[1,5]
β	10^{-28}	B^{UAV}	20MHz
δ_1, δ_2	1	B_i^u	0.5MHz
P_k	0.1 W	F^{UAV}	3 Ghz
P^{uR}	0.1 W	F^i	[10,12,14,16] GHz

toolbox in MATLAB. Specifically, the resource allocation scheme based on the SCA algorithm is shown in Algorithm .

V. SIMULATION RESULTS AND ANALYSIS

In this section, simulation results are given to evaluate the effectiveness of the proposed algorithm. We consider a UAV to assist 4 RSU on the ground for ground vehicles in an area of 1500m*1500m. The specific parameter settings are shown in Table 1.

Note that in order to explain the efficiency of our proposed scheme, some other benchmark schemes are designed as follows.

1) Only-UAV: In this case, the computational tasks of the vehicle are offloaded to the UAV only and are processed only at the UAV.
2) Only-RSU: In this case, the vehicle's computational tasks are completely offloaded to the RSU for processing through the UAV as a relay.
3) No-Priorit-UAV-RSU: In this case, the vehicle performs partial offloading. The computational tasks of the vehicle can be offloaded to the UAV and RSU for parallel processing. But in this case, the computational resources at the RSU side are not allocated according to the priority weights of the computational tasks.

Fig. 3 compares the system cost of the proposed SCA algorithm-based priority resource allocation scheme with the other three schemes when the UAV has different allocatable bandwidth resources, where the system cost is the weighted sum of the processing delay to complete the vehicle computing task and the energy consumption of the UAV. We observe that the system cost of all schemes shows a decreasing trend as the allocable bandwidth resources owned by the UAV gradually increase. This is because the increase in bandwidth resources leads to a decrease in the transmission delay of the

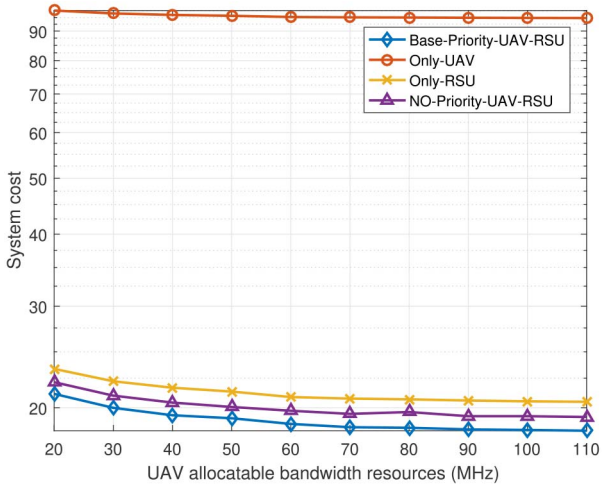

FIGURE 3. The relationship between bandwidth resources and system cost under different schemes.

vehicle uploading data to the UAV, which leads to a decrease in the system cost. Clearly, despite the increasing bandwidth resources, the system cost of the Only-UAV scheme is much higher than the other schemes. This is because each vehicle will receive significantly fewer computational resources if the vehicle only chooses to offload tasks to the UAV for processing where computational resources are limited. Since computational resources are proportional to processing latency, the reduction in computational resources will result in a larger system cost. In addition, we can see from the figure that the Base-Priority-UAV-RSU scheme proposed in this paper has the smallest system cost. Compared with the Only-UAV scheme and the Only-RSU scheme, the Base-Priority-UAV-RSU scheme can offload tasks to both UAV and RSU for parallel processing, so the task processing latency is greatly reduced, which results in a smaller system cost. Compared with the No-Priority-UAV-RSU scheme, the RSU in the Base-Priority-UAV-RSU scheme no longer divides the computing resources equally among the vehicle computing tasks, but allocates the resources dynamically according to the real-time demand of the computing tasks, so the system cost is smaller.

Fig. 4 represents the relationship between the total computing resources (computing power) of the UAV and the system cost. It is obvious in Fig. 4 that the system cost of the UAV-only solution decreases rapidly when the UAV computing resources gradually increase until 20 GHz. This is because, under this policy, all users offload their computing tasks to the UAV and are processed by the UAV only. And as the UAV computing resources increase, the processing latency of each task continues to decrease. As the UAV computing resources increase to 20 GHz, the rate of system cost reduction for the UAV-only scheme gradually moderates and overlaps with the priority-based and priority-free scheme. This is because as the computational resources consumed increase, although the processing latency decreases,

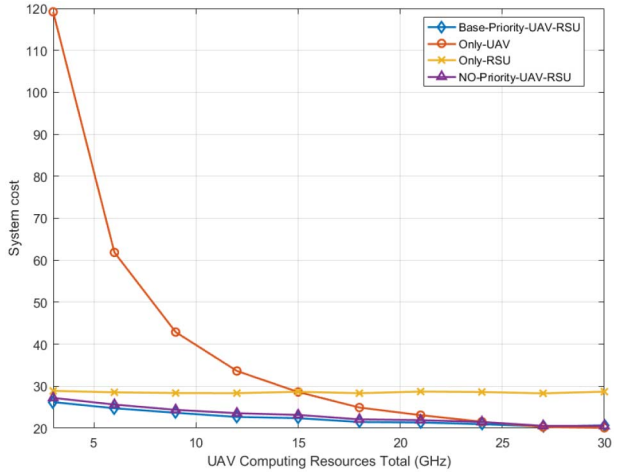

FIGURE 4. Relationship between calculation and system costs under different schemes.

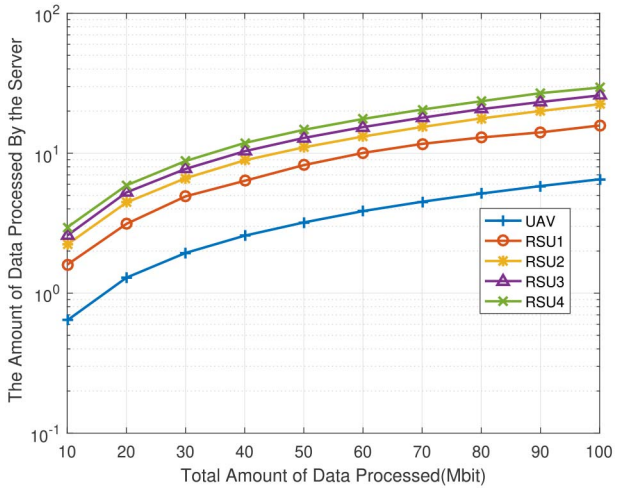

FIGURE 5. The amount of data processed by each server under different total amounts of data.

the UAV processing energy consumption gradually increases, which eventually affects the system cost to converge to a balance. In the RSU-only scenario, the UAV only acts as a forwarding relay to assist the user in offloading all computing tasks to the RSU server side for processing. Therefore the increase in UAV computing resources does not affect the system cost of this scheme. From the figure we can see that the two curves of the proposed scheme in this paper and the no-priority scheme nearly overlap. When the UAV computing resources increase, the amount of data processed at the UAV side increases and the amount of data processed at the RSU server side decreases. Therefore, the allocation of computing resources by priority at the RSU server side has a smaller impact on the system cost, resulting in a near overlap of these two curves.

From our system model, we know that each user's computational task is divided into 5 parts, which are offloaded to

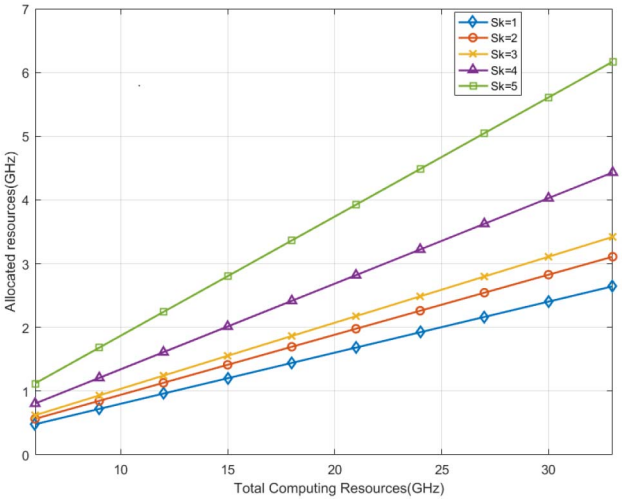

FIGURE 6. Relationship between computational resources obtained for data of different security levels and total computational resources of RSU1.

the UAV and 4 servers on the ground RSU side for processing. Fig. 5 represents the relationship between the amount of data processed on the server side for different RSU with different total amount of task data. As can be seen from the figure, the amount of data processed on the server side offloaded to the RSU is much larger than the amount of data processed offloaded to the UAV. Although the amount of data processed by each server has been increasing as the total amount of data to be processed has been increasing, the rate of increase in the amount of data processed by the servers at UAV and each RSU has been generally consistent. From this, we can conclude that when the resources are certain, the total amount of data processed by the server at the RSU side is much more than the total amount of data processed at the UAV side, i.e., the server processing delay at the RSU side has a greater impact on the system cost.

Fig. 6 represents the computational resources allocated to different levels of computational tasks as the computational resources of the server at RSU1 increase. When the computing resources of the server of RSU1 are less, the difference in computing resources available for all computing tasks are smaller. When the computing resources of the server of RSU1 increase, the computing resources allocated to different levels of computing tasks also gradually increase. Moreover, the gap in computational resources between the computational tasks assigned to higher-level computational tasks and lower-level computational tasks gradually increases. This is because different levels of computing tasks affect the priority coefficient of computing resource allocation. The higher the computational rank, the higher the priority weight, and the more computational resources are available. Therefore, the SCA-based priority resource allocation scheme proposed in this paper is able to dynamically divide the computing resources of the system according to the differential user requirements.

VI. CONCLUSION

This paper proposes an energy-efficient computation offloading strategy for multiple sensor data fusion in UAV Aided V2X Network supported by Integrated Sensing and Communication. We prioritize the data fusion tasks according to their importance, and minimize UAV energy consumption and data fusion task processing delay by jointly optimizing the offload decision and the computing resources of UAV and MEC. Since the joint optimization formulation problem is complex and non-convex, this paper proposes an optimization algorithm based on SCA. The algorithm transforms the above optimization problem into an approximate convex problem and then obtains the optimal solution through successive iterative. Simulation results show that the proposed strategy's performance is better than the baseline strategys in terms of reducing the UAV energy consumption and the weighted sum of data fusion task delay. In future work, we will fully exploit the maneuverability of UAVs. Rationally planning the flight trajectories and provide more reliable and stable data fusion processing services for ground vehicles.

REFERENCES

[1] J. Mu, Y. Gong, F. Zhang, Y. Cui, F. Zheng, and X. Jing, "Integrated sensing and communication-enabled predictive beamforming with deep learning in vehicular networks," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3301–3304, Oct. 2021.

[2] Y. Cui, X. Jing, and J. Mu, "Integrated sensing and communications via 5G NR waveform: Performance analysis," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2022, pp. 8747–8751.

[3] D. Teixeira and A. M. Pascoal, "Cooperative multiple formation control of autonomous marine vehicles," in *Proc. IEEE/OES Auton. Underwater Veh. Symp. (AUV)*, 2020, pp. 1–3.

[4] C. Lee and S.-L. Kim, "Most efficient sensor network protocol for a permanent natural disaster monitoring system," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 11776–11792, Aug. 2021.

[5] V. P. Kour and S. Arora, "Recent developments of the Internet of Things in agriculture: A survey," *IEEE Access*, vol. 8, pp. 129924–129957, 2020.

[6] L. Liu and M. Gruteser, "EdgeSharing: Edge assisted real-time localization and object sharing in urban streets," in *Proc. IEEE INFOCOM*, 2021, pp. 1–10.

[7] L. Zhao *et al.*, "Joint shareability and interference for multiple edge application deployment in mobile-edge computing environment," *IEEE Internet Things J.*, vol. 9, no. 3, pp. 1762–1774, Feb. 2022.

[8] K. Guo, R. Gao, W. Xia, and T. Q. S. Quek, "Online learning based computation offloading in MEC systems with communication and computation dynamics," *IEEE Trans. Commun.*, vol. 69, no. 2, pp. 1147–1162, Feb. 2021.

[9] W. Zhang, G. Zhang, and S. Mao, "Joint parallel offloading and load balancing for cooperative-MEC systems with delay constraints," *IEEE Trans. Veh. Technol.*, vol. 71, no. 4, pp. 4249–4263, Apr. 2022.

[10] M. Jiang, Y. Li, Q. Zhang, and J. Qin, "Joint position and time allocation optimization of UAV enabled time allocation optimization networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3806–3816, May 2019.

[11] Z. Yang, C. Pan, K. Wang, and M. Shikh-Bahaei, "Energy efficient resource allocation in UAV-enabled mobile edge computing networks," *IEEE Trans. Wireless*, vol. 18, no. 9, pp. 4576–4589, Sep. 2019.

[12] X. Cao, J. Xu, and R. Zhang, "Mobile edge computing for cellular-connected UAV: Computation offloading and trajectory optimization," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, 2018, pp. 1–5.

[13] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, and G. Y. Li, "Joint offloading and trajectory design for UAV-enabled mobile edge computing systems," *IEEE Internet Things Mag.*, vol. 6, no. 2, pp. 1879–1892, Apr. 2019.

[14] Y. Sun, Z. Chen, M. Tao, and H. Liu, "Bandwidth gain from mobile edge computing and caching in wireless multicast systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 3992–4007, Jun. 2020.

[15] Q. Zhang, Z. Chen, B. Xia, X. Jiang, and C. Xiong, "Design and optimization of edge computing for data fusion in V2I cooperative systems," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, 2020, pp. 466–471.

[16] B. Dai, F. Xu, Y. Cao, and Y. Xu, "Hybrid sensing data fusion of cooperative perception for autonomous driving with augmented vehicular reality," *IEEE Syst. J.*, vol. 15, no. 1, pp. 1413–1422, Mar. 2021.

[17] Y. Xu, T. Zhang, Y. Liu, D. Yang, L. Xiao, and M. Tao, "UAV-assisted MEC networks with aerial and ground cooperation," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 7712–7727, Dec. 2021.

[18] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, Sep. 2018.

[19] W. Mei, Q. Wu, and R. Zhang, "Cellular-connected UAV: Uplink association, power control and interference coordination," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5380–5393, Nov. 2019.

[20] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," *IEEE Trans. Commun.*, vol. 64, no. 12, pp. 4983–4996, Dec. 2016.

[21] L. Lin, X. Liao, H. Jin, and P. Li, "Computation offloading toward edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1584–1607, Aug. 2019.

[22] R. W. Saaty, "The analytic hierarchy process—What it is and how it is used," *Math. Model.*, vol. 9, nos. 3–5, pp. 161–176, 1987. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0270025587904738

[23] J. Ji, K. Zhu, C. Yi, and D. Niyato, "Energy consumption minimization in UAV-assisted mobile-edge computing systems: Joint resource allocation and trajectory design," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8570–8584, May 2021.

[24] T. Zhang, Y. Xu, J. Loo, D. Yang, and L. Xiao, "Joint computation and communication design for UAV-assisted mobile edge computing in IoT," *IEEE Trans Ind. Informat.*, vol. 16, no. 8, pp. 5505–5516, Aug. 2020.

[25] Y. Xu, T. Zhang, D. Yang, and L. Xiao, "UAV-assisted relaying and MEC networks: Resource allocation and 3D deployment," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, 2021, pp. 1–6.

[26] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization—Part I: Theory," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1929–1944, Apr. 2017.

[27] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in UAV-enabled mobile edge computing," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3147–3159, Apr. 2020.

QIAN LIU received the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2020.

From 2018 to 2020, she was a visiting Ph.D. student with the University of Victoria, Victoria, Canada. She is currently a Lecturer with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China. Her research interests include virtual cells, mobile edge computing, vehicular networks, and communication-sensing-computing integration.

HAIRONG LIANG is currently pursuing the M.S. degree with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China.

His research interest includes convex optimization and mobile edge computing.

RUI LUO is currently pursuing the M.S. degree with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China.

Her research interests include vehicular networks and mobile edge computing.

QILIE LIU received the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2020.

From 2018 to 2020, she was a visiting Ph.D. student with the University of Victoria, Victoria, Canada. She is a Lecturer with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China. Her research interests include virtual cells, mobile edge computing, vehicular networks, and communication-sensing-computing integration.