Joint Optimization of Security Strength and Resource Allocation for Computation Offloading in Vehicular Edge Computing

Huizi Xiao, Jun Zhao, Jie Feng, Lei Liu, Qingqi Pei, and Weisong Shi, *Fellow, IEEE*

***Abstract*—Vehicular Edge Computing (VEC) is a promising new paradigm that has attracted much attention in recent years, which can enhance the storage and computing capabilities of vehicular networks to provide users with low latency and high-quality services. Due to the open access and unreliable wireless channels, some appropriate security measures should be implemented in the VEC to ensure information security. However, the operation of the security mechanism dominates supererogatory computing resources, thus affecting the performance of VEC systems. The scarcity of computation and energy resources of the vehicles conflicts with the requirement of tasks for time delay and information security. In this paper, taking the driving velocity and position of the vehicles, the number of lanes, the model and density of the attackers, and security strength into consideration, we formulate a max-min optimization problem to jointly optimize offloading decision, transmit power, task computation frequency, encryption computation frequency, edge computation frequency, and block length to obtain optimal secure information capacity and local computation delay. The formulated optimization problem is a mixed integer nonlinear programming (MINLP), which is intractable. We apply the generalized benders decomposition (GBD)-based method to solve it. The simulation results show that our proposed algorithms have convergence and effectiveness and achieve fairness among vehicles on the road.**

***Index Terms*—Computation offloading, resource allocation, secure information capacity, vehicular edge computing.**

This work is supported by the National Key Research and Development Program of China under Grant 2021YFB2700600, the National Natural Science Foundation of China under Grants 62102297, 62001357, 62202005, 62132013, and 62102295, the Fundamental Research Funds for the Central Universities under Grants ZYTS23180 and ZYTS23178, the Key Research and Development Programs of Shaanxi under Grants 2022GY-437 and 2021ZDLGY06-03, and the Guangdong High Level Innovation Reaearch Institution Project (2021B0909050008). (*Corresponding authors: Jie Feng; Qingqi Pei*)

H. Xiao and Q. Pei are with State Key Laboratory of ISN, School of Telecommunication Engineering, Xidian University, Xi'an, Shaanxi 710071, China, and also with Engineering Research Center of Trusted Digital Economy, Universities of Shaanxi Province (email: huizi_xiao@stu.xidian.edu.cn, qqpei@mail.xidian.edu.cn).

J. Zhao is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore (email: junzhao@ntu.edu.sg).

J. Feng is with the State Key Laboratory of ISN, School of Telecommunication Engineering, Xidian University, Xian, Shaanxi 710071, China, and also with the Shaanxi Key Laboratory of Information Communication Network and Security, Xian University of Posts & Telecommunications, Xian, Shaanxi 710121, China (email: jiefengcl@163.com).

Lei Liu is with the Guangzhou Institute of Technology, Xidian University, Guangzhou 510555, China (email:leiliu@xidian.edu.cn).

W. Shi is with the Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA (email: weisong@udel.edu).

I. Introduction

With the rapid proliferation of the Internet of Things (IoT), billions of mobile and stationary devices have been connected to achieve real-time application services. However, the traditional cloud computing paradigm faces some significant challenges, such as high latency and jitter [1]. Edge computing is an emerging distributed computing paradigm that extends the concept of cloud computing to the edge of the network, which refers to migrating computation, communication, and storage resources closer to the end-users to process the massive amount of data and tasks. Low latency, reliability, high mobility, and geographically distributed users are the main characteristics of edge computing, making it a suitable solution to satisfy the challenges of vehicular networks. The integration of edge computing with vehicular networks is known as vehicular edge computing (VEC), where the resource-constrained vehicles offload latency-sensitive and computation-intensive tasks to edge servers. By integrating information, communication, storage, and intelligence technologies, VEC can extend the computation capability to the vehicular network edge as well as play an essential role in improving traffic efficiency and enhancing road safety [2].

VEC can provide flexible computation resources and application services on-demand, which need vehicles to transmit necessary data and task requirements, so computation offloading technology is vital. The vehicles can significantly reduce the burden of computing and routing to improve resource utilization by offloading tasks to edge servers along the road. However, due to the open access, dynamic network topology, and insecure wireless channels, there are maybe some security risks and privacy disclosures in the data transmission process during the offloading tasks of the vehicles. There may be potential attackers or adversary vehicles among the vehicles on the road. VEC is more vulnerable to threats and attacks because of the limited resources and the lack of centralized control compared with cellular networks [3]. There will definitely be security risks if data and tasks are transmitted in plain information without any cryptographic measures. Therefore, security mechanisms should be implemented in vehicular edge environments to provide appropriate confidentiality, integrity, authenticity, and more protection.

The pivotal issue of computation offloading in VEC is the decision-making mechanism. The vehicle must determine whether to offload the task to the edge server or compute locally. If it takes more time and resources to offload tasks

to the edge server than local execution, and the information security is threatened, it is not worth the gain. The scarcity of vehicle computing and wireless transmission resources also challenges the joint optimization of efficiency and security strength because the security schemes consume supererogatory computational resources and cause communication overhead. Due to the competition for limited computing and network resources, there is a contradiction between resource optimization and more robust security. For dynamic time-varying application scenarios such as VEC, driving velocity, the density of vehicles and attackers, and the number of lanes will all affect the security of task offloading. Hence, the offloading decision needs to be optimized jointly with resource allocation and security transmission under time delay and energy consumption constraints.

Many works have focused on resource allocation and computation offloading scheduling in VEC. Y. Cao et al. [4] illustrate the concept of edge computing enabled Internet-of-Vehicles and design a quality of experience (QoE)-based node selection strategy to choose a proper edge node to achieve a satisfying quality of experience on the whole. The paper [5] proposes a framework for edge computing on the road named autonomous vehicular edge to increase the computational capabilities of vehicles in a decentralized manner. Based on the Walrasian equilibrium, the paper [6] jointly analyses the resource allocation and computation offloading to find the best strategies for vehicles and VEC servers. A software-defined vehicular edge computing architecture introduced in [7] to assign a controller not only guides the vehicles' task offloading strategy but also determines the edge cloud resource allocation strategy to obtain optimum. Considering a three-layer VEC architecture, Z. Wang et al. [8] propose an online offloading scheduling and resource allocation algorithm to improve the system performance, which uses a game-theoretic online algorithm to solve the computation task offloading scheduling problem. There are also some works to solve the problem of an untrusted environment and information security in VEC. VECTrust [9] is a novel model to support the trusted resource allocation algorithms for scientific data-intensive workflows in VEC computing environments. A privacy-preserving vehicular edge computing (PP-VEC) system architecture is proposed in [10] to disturb the context information of the connected vehicles based on differential privacy technology before uploading it to the base station for offloading decisions to protect privacy. Only a small amount of works are devoted to the joint optimization of resource allocation and security. B. Mao et al. [11] propose an artificial intelligence-based adaptive security specification mechanism for 6G IoT networks where the devices are connected to cellular networks via different frequency bands. The paper [12] establishes a unified quality-of-service and security provisioning framework for wiretapping cognitive radio networks. A minimum optimization problem is formulated in [13] to weight the time delay and authentication security level simultaneously. However, the above-mentioned works did not consider this special scenario of the VEC environment for joint optimization of resource allocation and security. The papers [14] and [15] consider the utility and security simultaneously in vehicular networks. They only use the game theoretical approach to reach the Nash equilibrium, and there is no joint optimization of resources and security variables.

In the research mentioned above, there are solutions to resource optimization and computation offloading scheduling in VEC, and there are solutions that try to take information security into account. However, it is rarely considered that the particularity of the vehicular network jointly optimizes the resource allocation and security in the computation offloading under the constraints of time delay and energy consumption. In order to provide an idea to the above problems of computation offloading in vehicular edge computing, we jointly optimize offloading decision to decide whether the vehicle offload the tasks to the edge server, local resources to save the time and energy of vehicles, edge computation resource to balance the services provided by edge server, and block length to affect the security level of transmitting frames. The contributions of this paper are as follows

- We formulate a max-min optimization problem to jointly optimize offloading decision, transmit power, task computation frequency, encryption computation frequency, edge computation frequency, and block length to obtain optimal secure information capacity and local computation delay. The time delay constraint of the task takes into account the driving velocity and position of the vehicles.
- The proposed scheme makes the most suitable offloading decision based on the computing and communication resources of the vehicles and the computation capacity of the edge server, under the constraints of task execution delay and energy consumption. Hence, the overall performance of the vehicle set on the road is optimal. We consider the model and density of the attackers in the vehicle set into the formulated problem.
- The formulated optimization problem is a mixed integer nonlinear programming (MINLP), which is intractable. We apply the generalized benders decomposition (GBD)-based method to solve it. The problem can be decomposed into a primal problem and a master problem, which provide the upper bound and the lower bound of the original problem, respectively. These two problems can be solved separately to obtain the optimal solution to the original problem.
- The simulation results show that our proposed algorithms have well convergence and effectiveness as well as achieve fairness to the secure information capacity and local computation delay among vehicles. Meanwhile, the proposed schemes have a significant performance advantage compared with other schemes.

The remainder of this paper is structured as follows. We introduce the system description and formulate the max-min optimization problem in Section II. The solution procedure of the optimization problem is presented in Section III. Section IV performs the simulations and results analysis of the proposed algorithms. We conclude this paper in Section V.

II. System Description

This section describes the system scenario and security quantification for local computation in vehicles and compu-

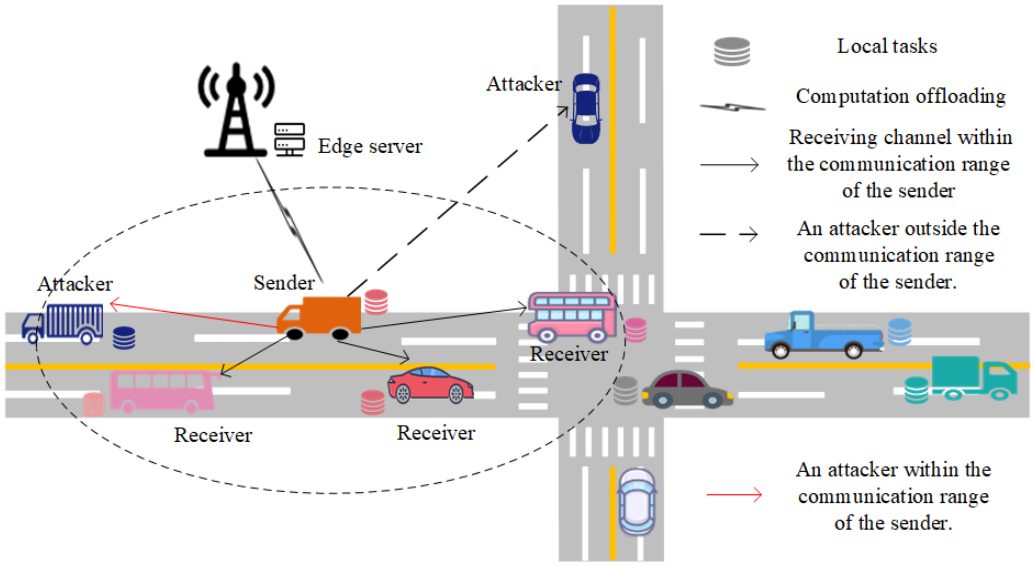

Fig. 1: The system scenario.

tation offloading to edge servers in VEC. Then, the whole process of local computation and offloading is modeled. Finally, the formal mathematical problem is formulated.

A. System Scenario

Every vehicle on the road has computational tasks that need to be handled, as shown in Fig.1, such as map recognition and navigation driving. The applications with strict latency and security requirements should be computed locally or offloaded to an edge server instead of being passed to a distant cloud center. However, offloading tasks out means more significant security risks and privacy breaches. Other vehicles on the road can access information within the communication range of the sending vehicle, so the appropriate encryptions are needed to protect the data in the offloading transmission. We consider an edge server covering a set of moving vehicles on a busy city road and can provide computation services for offloading tasks. Assume that there are I driving vehicles, which can be represented as the set $\mathcal{I} = \{1, 2, 3, ..., I\}$, and they connect with the edge server by TCP/IP protocol. There are sufficient reasons for vehicles to resort to the TCP/IP protocol, which is elaborated in the paper [16]. When sending vehicle offloads tasks, the data may be received by other malicious vehicles on the road, causing security risks, so it is necessary to protect the information using the mature and practical block cipher. Compared with other asymmetric encryption, symmetric encryption algorithms are simple, fast, efficient, and suitable for vehicular edge computing scenarios.

1) Computation Offloading in Vehicular Edge Computing: Many works [17] have used actual traffic data to be sure that the vehicle velocities obey a Gaussian distribution in the case of steady-state traffic conditions, such as free-flow and congestion, that have been accepted in vehicle traffic theory. In general, each driver is free to choose their comfortable driving speed according to the situation, so the velocities of different vehicles are independently and identically distributed (i.i.d.) [18]. In order to make each driving speed to the regular driving velocity range and thus avoid dealing with negative speeds or even values close to zero, we shall make the vehicle speed follow the truncated Gaussian distribution [19]. Since there are so many edge servers along the road in which the communication range is confined, and the vehicle velocity changes small in the short distance, each vehicle keeps its assigned speed v_i in an edge server coverage [20]. The minimum and maximum velocity of the truncated Gaussian distribution are represented v_{min} and v_{max}, separately. Therefore, we have the probability density function of the velocity,

$$f(v_i) = \begin{cases} \frac{2\exp(\frac{-(v_i-\mu)^2}{2\sigma^2})}{\sqrt{2\pi\sigma^2}\left(\mathrm{erf}\left(\frac{v_{max}-\mu}{\sqrt{2\sigma^2}}\right)-\mathrm{erf}\left(\frac{v_{min}-\mu}{\sqrt{2\sigma^2}}\right)\right)}, \\ \qquad\qquad v_{min} \le v_i \le v_{max}, \\ 0, \qquad\qquad \text{otherwise}, \end{cases} \tag{1}$$

where μ is the average velocity, σ^2 is the variance, and $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-\eta^2}\mathrm{d}\eta$ is the Gauss error function. By denoting the coverage length of the edge server on the road as L and representing the position of the vehicle entering the coverage as l_i, the remaining residence time of the driving vehicle in the current edge server coverage can be obtained

$$\tau_i = \frac{L - l_i}{v_i}. \tag{2}$$

The residence time can be used as the maximum delay for task execution to ensure integrity and timeliness.

Let p_i be the transmission energy per second of vehicle i, i.e., the transmit power. The channel transmission rate R_i depends on the p_i and connection state α_i of vehicle i as in [21], [22]

$$R_i = \alpha_i\sqrt{p_i}\ , \tag{3}$$

where α_i represents the state of supported vehicular connection and can be defined as

$$\alpha_i \triangleq \frac{K_0\sqrt{Z_i}}{RTT}\ , \tag{4}$$

where Z_i is the mobility function of the vehicle served by the considered TCP/IP mobile connection, which can be modeled by a time-correlated log-distributed sequence [23]. K_0 is a positive constant to capture the performance of the forward error correction-based error-recovery system in the physical layer. RTT is the round-trip-time of the wireless connection, which can be evaluated through Jacobson's formula.

2) Security Quantification with an Adversary Model: The security level of the encryption schemes can be measured by cryptographic scheme factors themselves, such as the difficulty of the mathematical problems and the length of the secret key. There may be different shortcut attacks against corresponding encryption algorithms, making it difficult to form a universal security quantification. On the other hand, in addition to being based solely on encryption schemes, the adversary's behavioral capabilities can be measured to represent security levels. Specifically, the ability of an attacker to crack a cipher of a certain block length is related to the probability mass function (PMF). The parameter "attacker strength" denoted by σ has the dimension of block length, and the probability of cracking a cipher of block length N is represented as $P_r(\sigma = N)$. An attacker with strength σ is able to crack any block cipher for length $\leq \sigma$ within the available time of the encrypted information with a cost less than its value. Hence, $P_r(\sigma = N)$ is also the probability that the frame would be cracked by the adversary, which leads to the leakage of the transmitted frame. The vulnerability of the message can be viewed as the probability of the frame leakage, which combined with the known probability theory then given by

$$\Phi = P_r(\sigma = N) = P_r(\sigma \geq N). \tag{5}$$

Let $\frac{1}{N_{max}-N_{min}}$ describe a vehicle adversary's strength, which applies a linear adversary strength model in [24]. N_{max} and N_{min} are the maximum and minimum block lengths available in the cryptosystem, respectively. This means that the probability of an adversary vehicle successfully attacking a block cipher of length N_i from vehicle i is uniformly distributed if it is one of the receivers within the communication range of the sending vehicle i. We define N_{min} and N_{max} frame length equal to 0 and N_f, respectively. Thus the vulnerability of the frames from $i \in \mathcal{I}$ is given by

$$\Phi_i = P_r(\sigma \geq N_i) = \frac{N_{max} - N_i}{N_{max} - N_{min}} = \frac{N_f - N_i}{N_f}, \tag{6}$$

which has an inverse relation to security level. The reason is that the lower the vulnerability of the frame means the higher the security level. Thus, the information security is defined as $1 - \Phi$. However, the data offloaded by vehicle i may also be received by adversary vehicles, and the information security level also depends on the number of adversary vehicles within the communication range. Therefore, the security level of a frame attacked by n_i^e attackers is $(1 - \Phi_i)^{n_i^e}$. The security strength of vehicle i offloading one block to the edge server can be obtained

$$S_i = (1 - \Phi_i)^{n_i^e} = \left(\frac{N_i}{N_f}\right)^{n_i^e}. \tag{7}$$

According to the paper [25], the vehicle density is denoted as $\gamma = \frac{N_l}{3v_{max}}$ by using the traffic model, where N_l is the number of lanes on the road. Thus, the number of attacker vehicles n_i^e is roughly given by

$$n_i^e = \pi d_i^2 \rho \gamma = \frac{\pi d_i^2 \rho N_l}{3v_{max}}, \tag{8}$$

where d_i is the coverage range radius of the sending vehicle i, and ρ vehicles/vehicle represents attackers' density. If the vehicle decides to offload its tasks, a node utility model in [14] is used to jointly optimize QoS and security, which utilizes measuring information [26] to formulate. The utility model of vehicle i can be expressed by the secure information capacity

$$U_i = \frac{R_i}{8N_i} \times \log_2(8N_i) \times S_i, \tag{9}$$

where $\frac{R_i}{8N_i}$ is the number of transmitted encryption blocks per second. The information capacity of each encryption block is $\log_2(8N_i)$, so the total information capacity of the transmitted blocks per second is induced as $\frac{R_i}{8N_i} \times \log_2(8N_i)$. S_i is the security strength per block given in (7). In this way, the above secure information capacity is created, which can be a part of the optimization goal.

B. Computation Mode Selection

There are two alternative computation modes available for vehicles. The offloading decision of vehicle i is denoted as $x_i \in \{0, 1\}$, where "1" represents that the vehicle decides to process the computation task locally and "0" expresses offloading the task to the edge server.

- Local computing: $x_i = 1$ means it is a local computation where the data does not leave the vehicles. So, the task does not need to be encrypted to ensure secure transmission, which requires processing the task directly with the on-board processors.
- Edge computing: $x_i = 0$ means that it is an offloading computation, and the data needs to be transferred to the edge server. Therefore, the vehicle needs to encrypt the data to ensure a certain transmission security level. Then the task is performed by the edge server. Compared to uplink transmission and encryption for the vehicle, backhaul and decryption at the edge server are negligible.

The specific details are described as follows

1) Local Computation in Vehicles: Let the data size of the task be D_i and set the average number of CPU cycles required to process one bit to q_i^c. Then the total CPU cycles to process the task are represented as $D_i q_i^c$. By denoting the local task computation frequency f_i^c, the time needed to process the task for local computation in vehicle i can be obtained

$$T_i^c = \frac{D_i q_i^c}{f_i^c}. \tag{10}$$

According to [27], the computation power of the on-board CPU to process task can be represented as $p_i^c = k_i {f_i^c}^3$. Therefore, the energy consumption can be given by

$$E_i^c = p_i^c T_i^c = k_i D_i q_i^c {f_i^c}^2, \tag{11}$$

where k_i is the effective switched capacitance based on the chip architecture [28]. By applying dynamic voltage and frequency scaling (DVFS) technology, we can adjust the computation frequency and supply voltage to minimize energy consumption.

2) Computation Offloading to Edge Server: If vehicle i decides to offload the task to the edge server for execution, an encryption scheme is required to protect the transmitted data, and applying block cipher does not increase the transmitted data size, so the data size still is D_i. Let the average number of CPU cycles required to encrypt one bit in the vehicle be q_i^{en} and process task one bit in the edge server be q_i^e. By denoting the encryption computation frequency f_i^{en} in the vehicle and the edge computation frequency f_i^e in the edge server, the time needed to encrypt is $\frac{D_i q_i^{en}}{f_i^{en}}$ and needed to process for the edge server is $\frac{D_i q_i^e}{f_i^e}$. The vehicle i requires to transmit data to the edge server, which the time it takes is $\frac{D_i}{R_i}$. Thus, the total time to offload and process task is

$$T_i^e = \underbrace{\frac{D_i q_i^{en}}{f_i^{en}}}_{\text{encryption delay}} + \underbrace{\frac{D_i}{\alpha_i \sqrt{p_i}}}_{\text{transmission delay}} + \underbrace{\frac{D_i q_i^e}{f_i^e}}_{\text{processing delay}}. \tag{12}$$

The computation power of the on-board CPU to encrypt data is $p_i^{en} = k_i f_i^{en3}$, just like section II-B1. Therefore, the total energy consumption of vehicle i to offload the task is

$$E_i^e = \underbrace{k_i D_i q_i^{en} f_i^{en2}}_{\text{encryption consumption}} + \underbrace{\frac{D_i \sqrt{p_i}}{\alpha_i}}_{\text{transmission consumption}}. \tag{13}$$

C. Problem Formulation

The vehicle i can choose the offloading decision $\boldsymbol{x} = (x_1, x_2, \cdots, x_I)$ based on its own and the edge server's capacity and security under the latency and energy consumption constraints of the task. Meanwhile, the transmit power $\boldsymbol{p} = (p_1, p_2, \cdots, p_I)$, the task computation frequency $\boldsymbol{f^c} = (f_1^c, f_2^c, \cdots, f_I^c)$, the encryption computation frequency $\boldsymbol{f^{en}} = (f_1^{en}, f_2^{en}, \cdots, f_I^{en})$ of vehicles, the edge computation frequency $\boldsymbol{f^e} = (f_1^e, f_2^e, \cdots, f_I^e)$ allocated to each vehicle by edge server, and the block length $\boldsymbol{N} = (N_1, N_2, \cdots, N_I)$ of the implemented block cipher can be jointly optimized to achieve the optimum. If the vehicle chooses local computation, i.e., $x_i = 1$, the latency can be the minimum optimization objective. Otherwise, the vehicle chooses to offload to the edge server, i.e., $x_i = 0$, the transmission security information capacity as the maximum optimization objective. Therefore, the maximum optimization objective can be written as follows

$$\mathcal{O}_i = (1 - x_i)\varpi_1 U_i - x_i \varpi_2 T_i^c, \tag{14}$$

where ϖ_1 and ϖ_2 are the scaling factors to merge the two parts into one objective formula to represent the target value within a reasonable range. However, there are I vehicles in the set on the road section. The whole system should be considered optimal, as well as the consumption between vehicles is supposed not to be significantly different, which needs to be balanced. We can achieve this by solving for the maximization of the minimum objective value in the vehicle set. Therefore, the final formulation can be mathematically expressed by

$$\begin{aligned} \max_{\boldsymbol{x},\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{N}} \ & \min_{i \in \mathcal{I}} \ \mathcal{O}_i \\ \text{s.t. } & (\mathrm{C}_1): x_i \in \{0,1\}, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_2): x_i T_i^c + (1 - x_i) T_i^e \le \tau_i, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_3): x_i E_i^c + (1 - x_i) E_i^e \le E_i, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_4): 0 \le f_i^c, f_i^{en} \le F_i^{loc}, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_5): \sum\nolimits_{i=1}^{I} f_i^e \le F^e, \\ & (\mathrm{C}_6): 0 \le p_i, 0 \le f_i^e, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_7): 0 \le N_i \le N_f, N_i \in \mathbb{N}^+, \ \forall i \in \mathcal{I}, \end{aligned} \tag{15}$$

where $\mathbb{N}^+$ represents the set of positive integers. E_i in (C_3), F_i^{loc} in (C_4), and F^e in (C_5) are the maximum energy consumption of vehicle i, the maximum local computation frequency for vehicle i, and the maximum computation frequency of the edge server, respectively. (C_1) indicates the vehicle decides whether to offload the task to the edge server. (C_2) represents that there is a maximum delay for task completion whether the task is executed locally or offloaded to the edge server. (C_6) restricts the edge computation frequency to be positive. (C_7) limits the transmission data block length.

First of all, to solve the problem efficiently, we represent the problem (15) in the epigraph form to transform the max-min problem into a maximum problem by introducing a new variable ζ. Then, the maximum problem can be transformed to a common and standard minimized form by letting $f_0 = -\zeta$ as follows

$$\begin{aligned} \min_{\zeta,\boldsymbol{x},\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{N}} \ & f_0 \\ \text{s.t. } & (\mathrm{C}_1), (\mathrm{C}_2), (\mathrm{C}_3), (\mathrm{C}_4), (\mathrm{C}_5), (\mathrm{C}_6), (\mathrm{C}_7), \\ & (\mathrm{C}_8): \zeta \le \mathcal{O}_i, \ \forall i \in \mathcal{I}. \end{aligned} \tag{16}$$

The formulated problem is intractable, which is a MINLP. A GBD-based method can be applied to solve it.

III. Solution of the Formulated Problem

The GBD method can be used to solve MINLP, i.e., extreme value problems that contain both integers and continuous variables, which uses the idea of a cutting plane to build an adequate solution representation. Specifically, the GBD method splits the original optimization problem into a master problem and a primal problem, which subtly lies in the introduction of complicating variables. After fixing the complicating variables, the remaining primal problem becomes relatively easy. Then, the extremum set of the master problem and the set of making the primal problem has a feasible solution are expressed appropriately using the cutting plane approach. We decompose the transformed problem (16) into a primal problem and a master problem, which are the convex optimization to obtain continuous variables and the mixed integer linear programming (MILP) to obtain discrete integer variables separately. The optimal value of the primal problem provides the upper bound of the original problem, while the

optimal value of the master problem provides the lower bound. The two problems iterate over each other until the method converges. The specific details are described as follows

A. Primal Problem

For the given integer variables $\boldsymbol{x}$ and $\boldsymbol{N}$, the remaining variables optimization is considered our primal problem, which is expressed as follows

$$\begin{aligned}\min_{\zeta,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e}}\ & f_0\\ \text{s.t. }& (\mathrm{C}_2),(\mathrm{C}_3),(\mathrm{C}_4),(\mathrm{C}_5),(\mathrm{C}_6),(\mathrm{C}_8).\end{aligned} \tag{17}$$

The GBD method requires the dual problem of the primal problem, which satisfies Slater's condition and strong duality, to solve for the optimal value. The problem is a convex one, as evidenced in Appendix A. We can represent the Lagrangian dual problem in (18) with zero dual gap to obtain the optimal solution of the primal problem by introducing the dual variable set $\boldsymbol{\Xi} = \{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\phi},\varphi,\boldsymbol{\psi}\}$. We have $\boldsymbol{\lambda} = \{\lambda_1,\lambda_2,\cdots,\lambda_I\} \succeq 0$, $\boldsymbol{\mu} = \{\mu_1,\mu_2,\cdots,\mu_I\} \succeq 0$, $\boldsymbol{\beta} = \{\beta_1,\beta_2,\cdots,\beta_I\} \succeq 0$, $\boldsymbol{\phi} = \{\phi_1,\phi_2,\cdots,\phi_I\} \succeq 0$, $\varphi \geq 0$, and $\boldsymbol{\psi} = \{\psi_1,\psi_2,\cdots,\psi_I\} \succeq 0$ corresponding to the constraints (C_2), (C_3), (C_4), (C_5), (C_8) in (17), respectively.

$$G(\boldsymbol{\Xi}) = \min_{\zeta,0\preceq\{\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e}\}} \mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi}), \tag{18}$$

where $\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})$ is the Lagrangian function represented in (19) as follows

$$\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi}) = f_0 + f_1(\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi}), \tag{19}$$

and

$$\begin{aligned}&f_1(\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})\\ &= \sum_{i=1}^{I}\lambda_i\Big(x_i\frac{D_iq_i^c}{f_i^c} + (1-x_i)\Big(\frac{D_iq_i^{en}}{f_i^{en}} + \frac{D_i}{\alpha_i\sqrt{p_i}} + \frac{D_iq_i^e}{f_i^e}\Big) - \tau_i\Big)\\ &+ \sum_{i=1}^{I}\mu_i\Big(x_ik_iD_iq_i^cf_i^{c2} + (1-x_i)\Big(k_iD_iq_i^{en}f_i^{en2} + \frac{D_i\sqrt{p_i}}{\alpha_i}\Big) - E_i\Big)\\ &+ \sum_{i=1}^{I}\beta_i\Big(f_i^c - F_i^{loc}\Big) + \sum_{i=1}^{I}\phi_i\Big(f_i^{en} - F_i^{loc}\Big) + \varphi\Big(\sum_{i=1}^{I}f_i^e - F^e\Big)\\ &+ \sum_{i=1}^{I}\psi_i\Big(\zeta - (1-x_i)\varpi_1\frac{\alpha_i\sqrt{p_i}}{8N_i}\times\log_2(8N_i)\times\Big(\frac{N_i}{N_f}\Big)^{n_i^e}\\ &\qquad + x_i\varpi_2\frac{D_iq_i^c}{f_i^c}\Big).\end{aligned} \tag{20}$$

1) Feasible Solution: When the primal problem (17) is feasible for the given offloading decision and block length, by applying the Karush-Kuhn-Tucker (KKT) conditions, we can obtain the optimal solution structure of the transmit power $\boldsymbol{p}$, the task computation frequency $\boldsymbol{f^c}$, the encryption computation frequency $\boldsymbol{f^{en}}$, and the edge computation frequency $\boldsymbol{f^e}$.

(i) The optimal solution structure of the transmit power

We take the partial derivatives of $\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})$ in (19) with respect to variable p_i and make it equal to 0.

$$\begin{aligned}&\frac{\partial\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})}{\partial p_i^*}\\ &= \Big(\mu_iD_i - \psi_i\varpi_1\alpha_i^2\frac{\log_2(8N_i)}{8N_i}\Big(\frac{N_i}{N_f}\Big)^{n_i^e}\Big)p_i - \lambda_iD_i = 0.\end{aligned} \tag{21}$$

So, we get the optimal structure of the transmit power as follows

$$p_i^* = \frac{(1-x_i)\lambda_iD_i}{\mu_iD_i - \psi_i\varpi_1\alpha_i^2\frac{\log_2(8N_i)}{8N_i}\Big(\frac{N_i}{N_f}\Big)^{n_i^e}}. \tag{22}$$

(ii) The optimal solution structure of the task computation frequency

We take the partial derivatives of $\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})$ in (19) with respect to variable f_i^c and make it equal to 0.

$$\begin{aligned}&\frac{\partial\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})}{\partial f_i^{c*}}\\ &= 2\mu_ik_iD_iq_i^cf_i^{c3} + \beta_if_i^{c2} - D_iq_i^c(\psi_i\varpi_2+\lambda_i) = 0.\end{aligned} \tag{23}$$

Let $\Delta = 3D_iq_i^c\sqrt[3]{\mu_i^2k_i^2(\psi_i\varpi_2+\lambda_i)}$, $Y_1 = \beta_i^3 - 2\Delta^3 + 3\mu_ik_iD_iq_i^c\sqrt{12D_iq_i^c(\psi_i\varpi_2+\lambda_i)(\Delta^3-\beta_i^3)}$, $Y_2 = \beta_i^3 - 2\Delta^3 - 3\mu_ik_iD_iq_i^c\sqrt{12D_iq_i^c(\psi_i\varpi_2+\lambda_i)(\Delta^3-\beta_i^3)}$, and $G = 1 - \frac{2\Delta^3}{\beta_i^3}$, we can get the optimal structure of the task computation frequency as follows when $x_i = 1$,

$$f_i^{c*} = \begin{cases}\frac{-\beta_i-(\sqrt[3]{Y_1}+\sqrt[3]{Y_2})}{6\mu_ik_iD_iq_i^c}, & \beta_i\in(0,-(\sqrt[3]{Y_1}+\sqrt[3]{Y_2})]\cap(0,\Delta),\\ \frac{\Delta^3}{6\mu_ik_iD_iq_i^c\beta_i^2}, & \beta_i = \Delta,\\ \frac{\beta_i(\cos\frac{\arccos G}{3}+\sqrt{3}\sin\frac{\arccos G}{3}-1)}{6\mu_ik_iD_iq_i^c}, & \beta_i > \Delta,\\ \text{no solution}, & \text{otherwise.}\end{cases} \tag{24}$$

(iii) The optimal solution structure of the encryption computation frequency

Similar to the task computation frequency, we can get the equation of the encryption computation frequency as follows

$$\begin{aligned}&\frac{\partial\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})}{\partial f_i^{en*}}\\ &= 2\mu_ik_iD_iq_i^{en}f_i^{en3} + \phi_if_i^{en2} - D_iq_i^{en}\lambda_i = 0.\end{aligned} \tag{25}$$

Let $\bar{\Delta} = 3D_iq_i^{en}\sqrt[3]{\mu_i^2k_i^2\lambda_i}$, $\bar{Y}_1 = \phi_i^3 - 2\bar{\Delta}^3 + 3\mu_ik_iD_iq_i^{en}\sqrt{12D_iq_i^{en}\lambda_i(\bar{\Delta}^3-\phi_i^3)}$, $\bar{Y}_2 = \phi_i^3 - 2\bar{\Delta}^3 - 3\mu_ik_iD_iq_i^{en}\sqrt{12D_iq_i^{en}\lambda_i(\bar{\Delta}^3-\phi_i^3)}$, and $\bar{G} = 1 - \frac{2\bar{\Delta}^3}{\phi_i^3}$, so the optimal structure of the encryption computation frequency can be obtained as follows when $x_i = 0$.

$$f_i^{en*} = \begin{cases}\frac{-\phi_i-(\sqrt[3]{\bar{Y}_1}+\sqrt[3]{\bar{Y}_2})}{6\mu_ik_iD_iq_i^{en}}, & \phi_i\in(0,-(\sqrt[3]{\bar{Y}_1}+\sqrt[3]{\bar{Y}_2})]\cap(0,\bar{\Delta}),\\ \frac{\bar{\Delta}^3}{6\mu_ik_iD_iq_i^{en}\phi_i^2}, & \phi_i = \bar{\Delta},\\ \frac{\phi_i(\cos\frac{\arccos\bar{G}}{3}+\sqrt{3}\sin\frac{\arccos\bar{G}}{3}-1)}{6\mu_ik_iD_iq_i^{en}}, & \phi_i > \bar{\Delta},\\ \text{no solution}, & \text{otherwise.}\end{cases} \tag{26}$$

(iv) The optimal solution structure of the edge computation frequency

Similar to the transmit power, we can get the equation of the edge computation frequency as follows

$$\frac{\partial\mathcal{L}(f_0,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi})}{\partial f_i^{e*}} = -\lambda_i(1-x_i)\frac{D_iq_i^e}{f_i^{e2}} + \varphi = 0. \tag{27}$$

Thus, the optimal structure of the edge computation frequency can be obtained as follows

$$f_i^{e*} = \sqrt{\frac{(1-x_i)\lambda_iD_iq_i^e}{\varphi}}. \tag{28}$$

Obviously, once we get the dual variables substituted into equations (22), (24), (26), and (28), we can find the optimal solution to the primal problem. The Lagrange dual variables set $\boldsymbol{\Xi}$ can be obtained by solving the following problem,

$$\begin{aligned} \max_{\boldsymbol{\Xi}} \quad & G(\boldsymbol{\Xi}) \\ \text{s.t. } & \boldsymbol{\Xi} = \{\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\phi}, \varphi, \boldsymbol{\psi}\} \succeq 0. \end{aligned} \tag{29}$$

The subgradient projection method can be applied to generate dual variables iteratively. The iterative equations are shown below.

$$\begin{aligned} \lambda_i(t+1) &= [\lambda_i(t) - m(t) \bigtriangledown \lambda_i(t)]^+, \\ \mu_i(t+1) &= [\mu_i(t) - n(t) \bigtriangledown \mu_i(t)]^+, \\ \beta_i(t+1) &= [\beta_i(t) - k(t) \bigtriangledown \beta_i(t)]^+, \\ \phi_i(t+1) &= [\phi_i(t) - i(t) \bigtriangledown \phi_i(t)]^+, \\ \varphi(t+1) &= [\varphi(t) - j(t) \bigtriangledown \varphi(t)]^+, \\ \psi_i(t+1) &= [\psi_i(t) - o(t) \bigtriangledown \psi_i(t)]^+, \end{aligned} \tag{30}$$

where $[h]^+ \triangleq \max\{0, h\}$, t is the subscript of the number of iterations, $m(t)$, $n(t)$, $k(t)$, $i(t)$, $j(t)$ and $o(t)$ are small positive step size. A set of subgradients of $G(\boldsymbol{\Xi})$ can be given by *Theorem 1* in paper [20], which can be represented as follows and proofed in Appendix B.

$$\begin{aligned} \bigtriangledown\lambda_i &= x_i \frac{D_i q_i^c}{f_i^{c*}} + (1-x_i)\Big(\frac{D_i q_i^{en}}{f_i^{en*}} + \frac{D_i}{\alpha_i\sqrt{p_i}^*} + \frac{D_i q_i^e}{f_i^{e*}}\Big) - \tau_i, \\ &\qquad \forall i \in \mathcal{I}, \\ \bigtriangledown\mu_i &= x_i k_i D_i q_i^c f_i^{c*2} + (1-x_i)\Big(k_i D_i q_i^{en} f_i^{en*2} + \frac{D_i\sqrt{p_i}^*}{\alpha_i}\Big) \\ &\qquad - E_i, \ \forall i \in \mathcal{I}, \\ \bigtriangledown\beta_i &= f_i^{c*} - F_i^{loc}, \ \forall i \in \mathcal{I}, \\ \bigtriangledown\phi_i &= f_i^{en*} - F_i^{loc}, \ \forall i \in \mathcal{I}, \\ \bigtriangledown\varphi &= \sum_{i=1}^{I} f_i^{e*} - F^e, \\ \bigtriangledown\psi_i &= \zeta - (1-x_i)\varpi_1 \frac{\alpha_i\sqrt{p_i}^*}{8N_i}\log_2(8N_i)\Big(\frac{N_i}{N_f}\Big)^{n_i^e} \\ &\qquad + x_i\varpi_2\frac{D_i q_i^c}{f_i^{c*}}, \ \forall i \in \mathcal{I}. \end{aligned} \tag{31}$$

The Lagrangian dual variables and the optimized solutions iterate over each other as described in Algorithm 1, and finally converge to the optimal solution of the primal problem. Hence, the optimality Benders cut can be added to the master problem at this iteration.

2) Infeasible Solution: There may not be a feasible solution to the primal problem (17). We formulate the following slack primal problem

$$\begin{aligned} \min_{\zeta, \hat{\boldsymbol{p}}, \hat{\boldsymbol{f^c}}, \hat{\boldsymbol{f^{en}}}, \hat{\boldsymbol{f^e}}, \hat{\boldsymbol{H}}, \hat{\boldsymbol{Q}}, \hat{\boldsymbol{Y}}} \quad & \sum_{i=1}^{I} \hat{H}_i + \hat{Q}_i + \hat{Y}_i \\ \text{s.t. } & (\hat{\mathrm{C}}_2): x_i T_i^c + (1-x_i)T_i^e \le \tau_i + \hat{H}_i, \ \forall i \in \mathcal{I}, \\ & (\hat{\mathrm{C}}_3): x_i E_i^c + (1-x_i)E_i^e \le E_i + \hat{Q}_i, \ \forall i \in \mathcal{I}, \\ & (\mathrm{C}_4), (\mathrm{C}_5), (\mathrm{C}_6), \\ & (\hat{\mathrm{C}}_8): \zeta \le \mathcal{O}_i + \hat{Y}_i, \ \forall i \in \mathcal{I}, \end{aligned} \tag{32}$$

Algorithm 1 Algorithm for Solving the Primal Problem

Initialization:

- Initialize the dual variables $\boldsymbol{\lambda}(0)$, $\boldsymbol{\mu}(0)$, $\boldsymbol{\beta}(0)$, $\boldsymbol{\phi}(0)$, $\varphi(0)$, $\boldsymbol{\psi}(0)$, maximum number of iterations t_{max} and the specified precision ϵ.
- Let $t = 0$.

Iteration:

1: **while** $t \le t_{max}$ **do**
2: Substitute the dual variables $\boldsymbol{\lambda}(t)$, $\boldsymbol{\mu}(t)$, $\boldsymbol{\beta}(t)$, $\boldsymbol{\phi}(t)$, $\varphi(t)$, $\boldsymbol{\psi}(t)$ into (22), (24), (26), and (28) to obtain $p_i(t)$, $f_i^c(t)$, $f_i^{en}(t)$, and $f_i^e(t)$, respectively.
3: Update new dual variables $\boldsymbol{\lambda}(t+1)$, $\boldsymbol{\mu}(t+1)$, $\boldsymbol{\beta}(t+1)$, $\boldsymbol{\phi}(t+1)$, $\varphi(t+1)$, $\boldsymbol{\psi}(t+1)$ by using (30) and (31), according to the new $p_i(t)$, $f_i^c(t)$, $f_i^{en}(t)$, and $f_i^e(t)$.
4: **if** $||\boldsymbol{\lambda}(t+1) - \boldsymbol{\lambda}(t)|| < \epsilon$, $||\boldsymbol{\mu}(t+1) - \boldsymbol{\mu}(t)|| < \epsilon$, $||\boldsymbol{\beta}(t+1) - \boldsymbol{\beta}(t)|| < \epsilon$, $||\boldsymbol{\phi}(t+1) - \boldsymbol{\phi}(t)|| < \epsilon$, $||\varphi(t+1) - \varphi(t)|| < \epsilon$, and $||\boldsymbol{\psi}(t+1) - \boldsymbol{\psi}(t)|| < \epsilon$ **then**
5: $p_i^* = p_i(t)$, $f_i^{c*} = f_i^c(t)$, $f_i^{en*} = f_i^{en}(t)$, and $f_i^{e*} = f_i^e(t)$.
6: break.
7: **else**
8: $t = t + 1$.
9: **end if**
10: **end while**

Output: $\boldsymbol{p}^*$, $\boldsymbol{f^{c*}}$, $\boldsymbol{f^{en*}}$, $\boldsymbol{f^{e*}}$.

where $\hat{\boldsymbol{H}} = \{\hat{H}_1, \hat{H}_2, \cdots, \hat{H}_I\}$, $\hat{\boldsymbol{Q}} = \{\hat{Q}_1, \hat{Q}_2, \cdots, \hat{Q}_I\}$, and $\hat{\boldsymbol{Y}} = \{\hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_I\}$ are the slack variables with respect to the constraints (C_2), (C_3), and (C_8) in primal problem (17), respectively. The introduced variables are linear and do not affect the convexity of the original primal problem. So, we can still use a similar method to obtain the optimal solution of problem (32). By introducing the new dual variables set $\hat{\boldsymbol{\Xi}} = \{\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\phi}}, \hat{\varphi}, \hat{\boldsymbol{\psi}}\}$, the Lagrangian function of the new problem can be expressed as

$$\begin{aligned} &\hat{\mathcal{L}}(\zeta, \hat{\boldsymbol{p}}, \hat{\boldsymbol{f^c}}, \hat{\boldsymbol{f^{en}}}, \hat{\boldsymbol{f^e}}, \hat{\boldsymbol{H}}, \hat{\boldsymbol{Q}}, \hat{\boldsymbol{Y}}, \hat{\boldsymbol{\Xi}}) = \sum_{i=1}^{I}(\hat{H}_i + \hat{Q}_i + \hat{Y}_i) + \\ &\sum_{i=1}^{I}\hat{\lambda}_i\Big(x_i\frac{D_i q_i^c}{\hat{f}_i^c} + (1-x_i)\Big(\frac{D_i q_i^{en}}{f_i^{\hat{en}}} + \frac{D_i}{\alpha_i\sqrt{\hat{p}_i}} + \frac{D_i q_i^e}{\hat{f}_i^e}\Big) - \tau_i - \hat{H}_i\Big) \\ &+ \sum_{i=1}^{I}\hat{\mu}_i\Big(x_i k_i D_i q_i^c \hat{f}_i^{c\,2} + (1-x_i)\Big(k_i D_i q_i^{en} f_i^{\hat{en}\,2} + \frac{D_i\sqrt{\hat{p}_i}}{\alpha_i}\Big) \\ &- E_i - \hat{Q}_i\Big) + \sum_{i=1}^{I}\hat{\beta}_i\Big(\hat{f}_i^c - F_i^{loc}\Big) + \sum_{i=1}^{I}\hat{\phi}_i\Big(f_i^{\hat{en}} - F_i^{loc}\Big) \\ &+ \hat{\varphi}\Big(\sum_{i=1}^{I}\hat{f}_i^e - F^e\Big) + \sum_{i=1}^{I}\hat{\psi}_i\Big(\zeta - (1-x_i)\varpi_1\frac{\alpha_i\sqrt{\hat{p}_i}}{8N_i} \times \log_2(8N_i) \\ &\times \Big(\frac{N_i}{N_f}\Big)^{n_i^e} + x_i\varpi_2\frac{D_i q_i^c}{\hat{f}_i^c} - \hat{Y}_i\Big). \end{aligned} \tag{33}$$

For the given slack variables $\hat{\boldsymbol{H}}$, $\hat{\boldsymbol{Q}}$, and $\hat{\boldsymbol{Y}}$, after obtaining the optimal solution (22), (24), (26), (28) and iteration steps (30), (31) by using the same approach in Section III-A1, the variables $\tilde{\boldsymbol{p}}^*$, $\tilde{\boldsymbol{f^c}}^*$, $\tilde{\boldsymbol{f^{en}}}^*$, $\tilde{\boldsymbol{f^e}}^*$ can be acquired by calling the variant of Algorithm 1. The variant of Algorithm 1 refers

to replacing $(\tilde{22})$, $(\tilde{24})$, $(\tilde{26})$, $(\tilde{28})$, $(\tilde{30})$, and $(\tilde{31})$ with the corresponding (22), (24), (26), (28), (30), and (31) in Algorithm 1, and the rest of the logic and flow is the same. Then, we obtain the new slack variables given by

$$\begin{aligned}\hat{H}_i &= x_i T_i^c + (1-x_i)T_i^e - \tau_i,\\ \hat{Q}_i &= x_i E_i^c + (1-x_i)E_i^e - E_i,\\ \hat{Y}_i &= \zeta - \mathcal{O}_i.\end{aligned} \tag{34}$$

The idea of the block coordinate descent algorithm can be applied here to get the final optimal solution $\hat{\boldsymbol{p}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{c}}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{en}}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{e}}}^*$ of the slack problem (32) by calling Algorithm 2.

Algorithm 2 Algorithm for the Infeasible Solution of the Primal Problem

Initialization:

- Initialize the slack variable $\hat{\boldsymbol{H}}(0)$, $\hat{\boldsymbol{Q}}(0)$, and $\hat{\boldsymbol{Y}}(0)$.
- Set the maximum number of iterations l_{max} and the specified precision ε.
- Let $l = 1$.

Allocate the variables $\tilde{\boldsymbol{p}}^*(0)$, $\tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(0)$, $\tilde{\boldsymbol{f}^{\boldsymbol{en}}}^*(0)$, and $\tilde{\boldsymbol{f}^{\boldsymbol{e}}}^*(0)$ by calling the variant of Algorithm 1 based on $\hat{\boldsymbol{H}}(0)$, $\hat{\boldsymbol{Q}}(0)$, and $\hat{\boldsymbol{Y}}(0)$.
Substitute $\tilde{\boldsymbol{p}}^*(0)$, $\tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(0)$, $\tilde{\boldsymbol{f}^{\boldsymbol{en}}}^*(0)$, and $\tilde{\boldsymbol{f}^{\boldsymbol{e}}}^*(0)$ into (34) to obtain the new slack variables $\hat{\boldsymbol{H}}(1)$, $\hat{\boldsymbol{Q}}(1)$, and $\hat{\boldsymbol{Y}}(1)$.

Iteration:

while $l \le l_{max}$ **do**
  Compute $\tilde{\boldsymbol{p}}^*(l)$, $\tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(l)$, $\tilde{\boldsymbol{f}^{\boldsymbol{en}}}^*(l)$, and $\tilde{\boldsymbol{f}^{\boldsymbol{e}}}^*(l)$ by calling the variant of Algorithm 1 based on $\hat{\boldsymbol{H}}(l)$, $\hat{\boldsymbol{Q}}(l)$, and $\hat{\boldsymbol{Y}}(l)$.
  Substitute $\tilde{\boldsymbol{p}}^*(l)$, $\tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(l)$, $\tilde{\boldsymbol{f}^{\boldsymbol{en}}}^*(l)$, and $\tilde{\boldsymbol{f}^{\boldsymbol{e}}}^*(l)$ into (34) to obtain $\hat{\boldsymbol{H}}(l+1)$, $\hat{\boldsymbol{Q}}(l+1)$, and $\hat{\boldsymbol{Y}}(l+1)$.
  if $|\hat{\boldsymbol{H}}(l+1) - \hat{\boldsymbol{H}}(l)| \le \varepsilon$, $|\hat{\boldsymbol{Q}}(l+1) - \hat{\boldsymbol{Q}}(l)| \le \varepsilon$, and $|\hat{\boldsymbol{Y}}(l+1) - \hat{\boldsymbol{Y}}(l)| \le \varepsilon$ **then**
    $\hat{\boldsymbol{p}}^* = \tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(l)$, $\hat{\boldsymbol{f}^{\boldsymbol{c}}}^* = \tilde{\boldsymbol{f}^{\boldsymbol{c}}}^*(l)$, $\hat{\boldsymbol{f}^{\boldsymbol{en}}}^* = \tilde{\boldsymbol{f}^{\boldsymbol{en}}}^*(l)$, and $\hat{\boldsymbol{f}^{\boldsymbol{e}}}^* = \tilde{\boldsymbol{f}^{\boldsymbol{e}}}^*(l)$
    break.
  end if
  $l = l + 1$.
end while

Output: $\hat{\boldsymbol{p}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{c}}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{en}}}^*$, $\hat{\boldsymbol{f}^{\boldsymbol{e}}}^*$.

Therefore, the optimal solution of the slack primal problem, which is an infeasible solution to the original primal problem, can form a feasibility Benders cut added to the master problem at this iteration.

B. Master Problem

Depending on whether the primal problem (17) is feasible or not, we denote the number of feasible solutions as K^f and the number of infeasible solutions as K^I. They add optimality cuts and feasibility cuts to the master problem, respectively. Therefore, we can represent the master problem as follows

$$\begin{aligned}\min_{\boldsymbol{x},\boldsymbol{N},y_0}\ & y_0\\ \text{s.t. }& (\mathrm{C}_1),(\mathrm{C}_7),\\ & \mathscr{L}^1(\boldsymbol{x},\boldsymbol{N},\boldsymbol{\Xi}^{k^f}) \le y_0,\ k^f = 1,2,\cdots,K^f,\\ & \mathscr{L}^2(\boldsymbol{x},\boldsymbol{N},\hat{\boldsymbol{\Xi}}^{k^I}) \le 0,\quad k^I = 1,2,\cdots,K^I,\end{aligned} \tag{35}$$

where

$$\mathscr{L}^1(\boldsymbol{x},\boldsymbol{N},\boldsymbol{\Xi}^{k^f}) = \min_{\zeta,\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e}} f_0 + f_1(\boldsymbol{p},\boldsymbol{f^c},\boldsymbol{f^{en}},\boldsymbol{f^e},\boldsymbol{\Xi}^{k^f}), \tag{36}$$

and

$$\mathscr{L}^2(\boldsymbol{x},\boldsymbol{N},\hat{\boldsymbol{\Xi}}^{k^I}) = \min_{\hat{\boldsymbol{p}},\hat{\boldsymbol{f^c}},\hat{\boldsymbol{f^{en}}},\hat{\boldsymbol{f^e}}} f_1(\hat{\boldsymbol{p}},\hat{\boldsymbol{f^c}},\hat{\boldsymbol{f^{en}}},\hat{\boldsymbol{f^e}},\hat{\boldsymbol{\Xi}}^{k^I}). \tag{37}$$

The variables $\boldsymbol{p}$, $\boldsymbol{f^c}$, $\boldsymbol{f^{en}}$, $\boldsymbol{f^e}$, $\boldsymbol{\Xi}^{k^f}$ in (36) can derive from the primal solution and its dual solution to problem (17). As well as the variables $\hat{\boldsymbol{p}}$, $\hat{\boldsymbol{f^c}}$, $\hat{\boldsymbol{f^{en}}}$, $\hat{\boldsymbol{f^e}}$, $\hat{\boldsymbol{\Xi}}^{k^I}$ in (37) can derive from the slack primal solution and its dual solution to problem (32). The variables to be solved in the master problem (35) are $\boldsymbol{x}$ and $\boldsymbol{N}$. By analyzing the master problem, we know that solving for $\boldsymbol{N}$ in the master problem is equivalent to solving for $\boldsymbol{N}$ under constraint (C_7) when the following function takes a maximum value.

$$f(N_i) = (1-x_i)\varpi_1 \frac{\alpha_i\sqrt{p_i}}{8N_i}\log_2(8N_i)\Big(\frac{N_i}{N_f}\Big)^{n_i^e}. \tag{38}$$

The procedure for finding the optimal solution of function $f(N_i)$ is shown in Appendix C. So, if $x_i = 0$, we can obtain the optimal block length as follows

$$N_i{}^* = \begin{cases} \mathrm{round}(2^{\frac{1}{(1-n_i^e)\ln 2}-3}), & 0 \le n_i^e < 1,\\ N_f, & n_i^e \ge 1,\end{cases} \tag{39}$$

where $\mathrm{round}(\cdot)$ indicates rounding. The optimal solution is consistent with common sense. When an adversary vehicle may be present in the communication range of the sending vehicle, the sender needs to determine the transmission block length based on the probability of adversary presence to ensure a comparable level of security. When there is definitely one or more adversaries in the communication range of the sending vehicle, the sending vehicle will adjust its transmission block length to the maximum to try its best to ensure security strength.

The only variable that needs to be determined is $\boldsymbol{x}$ up to now. When there is a feasible solution to the primal problem (17) for this iteration, we substitute $x_i = 1$ and $x_i = 0$ for each vehicle i into function $\mathscr{L}^1(\boldsymbol{x},\boldsymbol{N},\boldsymbol{\Xi}^{k^f})$ to obtain the function values $\mathscr{L}^1_{i0}$ and $\mathscr{L}^1_{i1}$, then perform a simple comparison. If $\mathscr{L}^1_{i0} \le \mathscr{L}^1_{i1}$, x_i is set to 0. Otherwise, x_i is set to 1. When there is an infeasible solution to the primal problem (17) for this iteration, we substitute $x_i = 1$ and $x_i = 0$ for each vehicle i into function $\mathscr{L}^2(\boldsymbol{x},\boldsymbol{N},\hat{\boldsymbol{\Xi}}^{k^I})$ to obtain the function values $\mathscr{L}^2_{i0}$ and $\mathscr{L}^2_{i1}$, then perform a simple comparison. If $\mathscr{L}^2_{i0} \le \mathscr{L}^2_{i1}$, x_i is set to 0. Otherwise, x_i is set to 1. With the above analysis and equation (39), we obtain the optimal solution of the master problem (35).

Algorithm 3 The Generalized Benders Decomposition Method to Solve Problem (16)

Initialization:

- Initialize the integer variables $\boldsymbol{x}(0)$ and $\boldsymbol{N}(0)$.
- Set the maximum number of iterations m_{max}, the specified precision ι.
- Let $m = 0$.

Set $LB(0) = -\infty$, $UB(0) = +\infty$, $\varrho = \iota$.

Iteration:

while $m \leq m_{max}$ and $\varrho \geq \iota$ **do**
  Solve the primal problem (17) for the given $\boldsymbol{x}(m)$ and $\boldsymbol{N}(m)$.
  if the primal problem (17) is feasible **then**
    Obtain the $\boldsymbol{p}(m)$, $\boldsymbol{f^c}(m)$, $\boldsymbol{f^{en}}(m)$, $\boldsymbol{f^e}(m)$, $\boldsymbol{\Xi}(m)$, and $f_0(m)$.
    Update $UB(m+1) = \min\{UB(m), f_0(m)\}$.
    if $UB(m+1) = f_0(m)$ **then**
      $\boldsymbol{p}^{'} = \boldsymbol{p}(m)$, $\boldsymbol{f^{c}}^{'} = \boldsymbol{f^c}(m)$, $\boldsymbol{f^{en}}^{'} = \boldsymbol{f^{en}}(m)$, $\boldsymbol{f^{e}}^{'} = \boldsymbol{f^e}(m)$, $\boldsymbol{x}^{'} = \boldsymbol{x}(m)$ and $\boldsymbol{N}^{'} = \boldsymbol{N}(m)$.
    end if
  else
    Solve the slack primal problem (32) to obtain $\hat{\boldsymbol{p}}(m)$, $\hat{\boldsymbol{f^c}}(m)$, $\hat{\boldsymbol{f^{en}}}(m)$, $\hat{\boldsymbol{f^e}}(m)$, and $\hat{\boldsymbol{\Xi}}(m)$.
  end if
  Solve the master problem (35) to obtain $\boldsymbol{x}(m+1)$, $\boldsymbol{N}(m+1)$ and $y_0(m+1)$.
  Update $LB(m+1) = y_0(m+1)$.
  Update $\varrho = |\frac{UB(m+1)-LB(m+1)}{LB(m+1)}|$.
  $m = m + 1$.
end while

Output: $\boldsymbol{p}^* = \boldsymbol{p}^{'}$, $\boldsymbol{f^{c*}} = \boldsymbol{f^{c}}^{'}$, $\boldsymbol{f^{en*}} = \boldsymbol{f^{en}}^{'}$, $\boldsymbol{f^{e*}} = \boldsymbol{f^{e}}^{'}$, $\boldsymbol{x}^* = \boldsymbol{x}^{'}$, $\boldsymbol{N}^* = \boldsymbol{N}^{'}$.

C. Generalized Benders Decomposition Algorithm

The optimal value obtained in the primal problem (17) is the performance upper bound of the problem (16), which is represented as UB. And the optimal value obtained in the master problem (35) is the performance lower bound of the problem (16), which is represented as LB. The optimality Benders cuts and feasibility Benders cuts are continuously added to the master problem with the number of iterations. The search area for the optimal global solution gradually decreases. The generalized benders decomposition method to solve the problem (16) is expressed in Algorithm 3.

The optimal value of the problem (16) can be obtained when the gap between UB and LB reaches a preset threshold ι or the number of iterations reaches a maximum m_{max}. The optimal value of the maximum problem can be recovered from the optimal value of the problem (16) by $\zeta = -f_0$.

IV. Simulation Results and Analysis

This section performs the simulation of the algorithms and the analysis of the experimental results. Firstly, we introduce the environment and parameter settings for the experiments. Then, the convergence and feasibility of the algorithms are verified. Finally, we perform the performance analysis of the algorithms.

TABLE I: Parameter settings in the simulation

Parameter	Meaning	Value
v_{min}/v_{max}	Minimum / Maximum Speed	[2, 24] m/s [20]
μ	Mean Speed	13 m/s [20]
σ	Standard Deviation of Speed	5 [20]
L	Coverage Diameter of Edge Server	100 m
α_i	State of Vehicular Connection	[3, 6]$\times 10^6$
N_f	Maximum Block Length	64
ρ	Density of Attackers	10^{-4} vehicles/vehicle [14]
N_l	Number of Lanes	8
D_i	Data Size	[100, 900] KB
q_i^c	Processing Density of Task	600 cycles/bit
k_i	Effective Switched Capacitance	10^{-27} [29]
q_i^{en}	Processing Density of Encryption	90 cycles/bit
q_i^e	Processing Density of Edge	100 cycles/bit
ϖ_1, ϖ_2	The Weighted Values	[0,1] $\times 10^{-5}$, [0,1]
E_i	Maximum Energy of the Vehicle	0.4 J
F_i^{loc}	Maximum Vehicular Computation Frequency	2.0 GHz [30]
F^e	Maximum Edge Computation Frequency	3.0 GHz

A. Simulation Settings

We consider that there are currently 8 vehicles under the coverage of an edge server at the roadside. Some task data need to be processed in each vehicle. The vehicle can process the task locally or offload it to the edge server. However, if the vehicle decides to offload the task to the edge server will need to upload data. Some vehicles in the communication range of the sending vehicle can also receive this information. If there is a malicious vehicle in the receiver, the information will suffer security risks. Therefore, the sending vehicle needs to encrypt its data and decide the encryption block length according to the condition and capacity of the malicious vehicles to appear. Thus, in addition to some task and channel attributes, there are some parameters of traffic conditions and malicious vehicles that are also listed in Table I.

We jointly optimized some variables and randomly selected the remaining variables to highlight the proposed scheme's advantages. The experiments set the following schemes

- RTCF [20]: The scheme randomly chooses task computation frequency. The remaining variables are optimized, such as encryption computation frequency, transmit power, edge computation frequency, transmission block length, and vehicular offloading decision.
- RCOR [31]: The scheme randomly chooses computation offloading resources, such as encryption computation frequency, transmit power, and edge computation frequency. The remaining variables are optimized, such as task computation frequency, transmission block length, and vehicular offloading decision.
- RTBL [32]: The scheme randomly chooses transmission block length. The remaining variables are optimized, such as transmit power, task computation frequency, encryption computation frequency, edge computation frequency, and vehicular offloading decision.

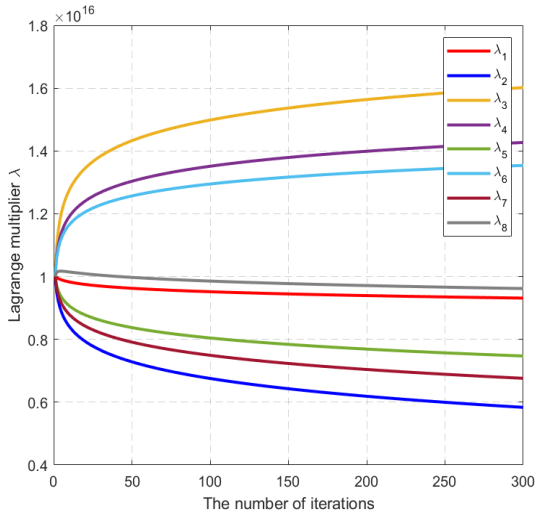

Fig. 2: Convergence of Algorithm 1 in feasible solution.

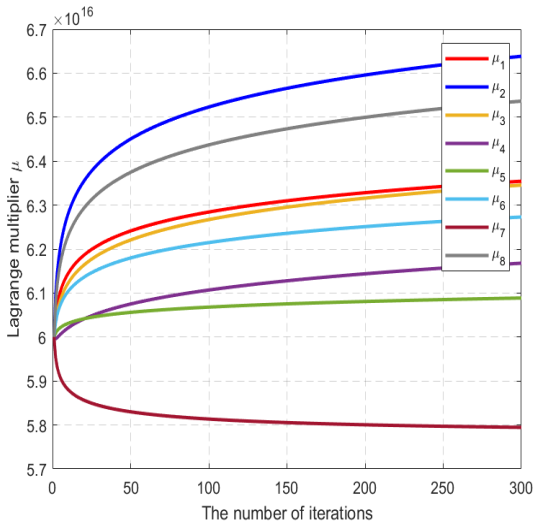

Fig. 3: Convergence of the variant of Algorithm 1 in infeasible solution.

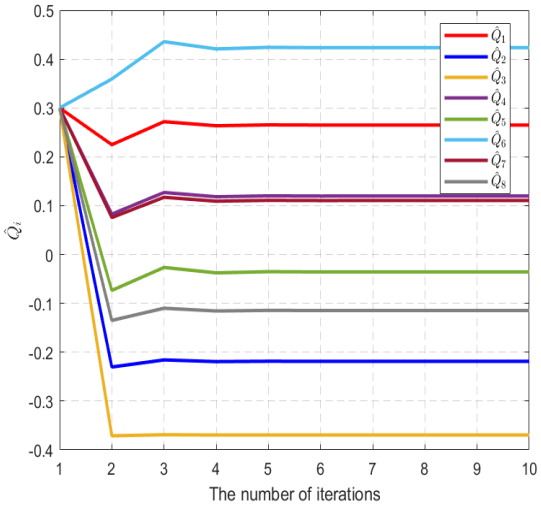

Fig. 4: Convergence of Algorithm 2 in infeasible solution.

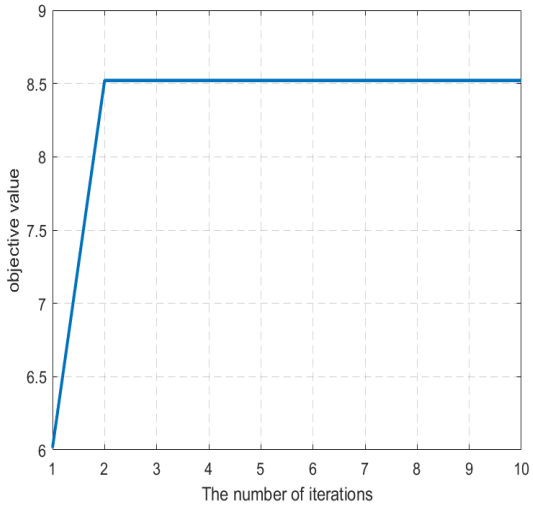

Fig. 5: The objective value of Algorithm 3 along with the iterations.

- RVOD [33]: The scheme randomly chooses vehicular offloading decision. The remaining variables are optimized, such as transmit power, task computation frequency, encryption computation frequency, edge computation frequency, and transmission block length.

B. Convergence of the Proposed Algorithms

The curves of the Lagrange multipliers λ_i in Algorithm 1 with the number of iterations are shown in Fig. 2. It can be seen that the feasible solutions for all I vehicles can converge simultaneously, which also indicates the effectiveness and convergence of the Algorithm 1. For the infeasible solution of the primal problem, its internal lagrangian subgradient projection method is implemented by calling the variant of Algorithm 1. The curves of Lagrange multipliers μ_i for all I vehicles with the number of iterations are plotted in Fig. 3, which exhibits a good convergence as the basis for obtaining stable infeasible solutions. Algorithm 2 obtains an infeasible solution to the primal problem by cyclically calling the variant of Algorithm 1 and computing new values of the slack variables $\hat{\boldsymbol{H}}$, $\hat{\boldsymbol{Q}}$, and $\hat{\boldsymbol{Y}}$. Fig. 4 shows the trend of the slack variables $\hat{Q}_i$ with the number of iterations. The figure demonstrates the fast convergence of the infeasible solution for the whole I vehicles system. Since I vehicles have different parameters such as speed, position, connection status, and communication range, there will be a situation where some vehicles satisfy a single constraint while others do not, so the slack variables $\hat{Q}_i$ of some vehicles are negative and others are positive. But no matter whether positive or negative, their slack variables eventually converge to stable, indicating that the system as a whole converges to infeasible solutions. Fig. 5 shows the trend of the objective value in Algorithm 3 with the number of iterations, indicating the fast convergence of the generalized benders decomposition method for solving problem (16). As the iterations proceed, the solution does not necessarily jump across between the feasible and infeasible solutions, and it is

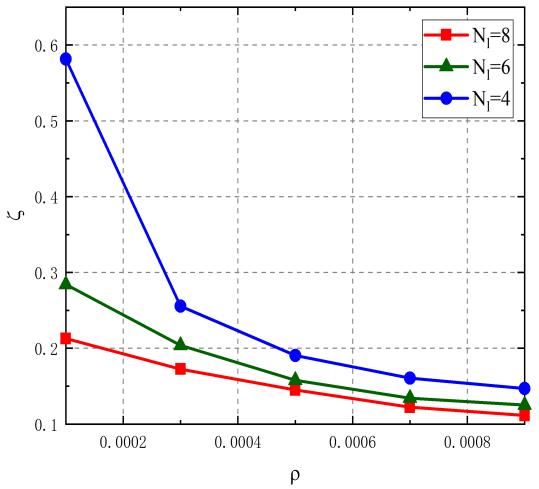

Fig. 6: ζ under the different density of attackers ρ and number of lanes N_l.

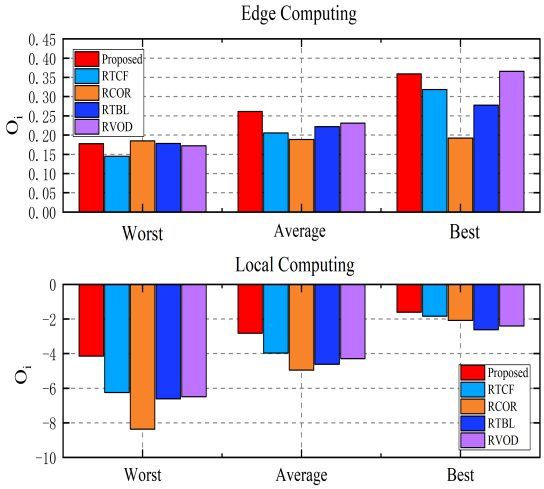

Fig. 7: Comparisons of the average objective value, the worst objective value, and the best objective value.

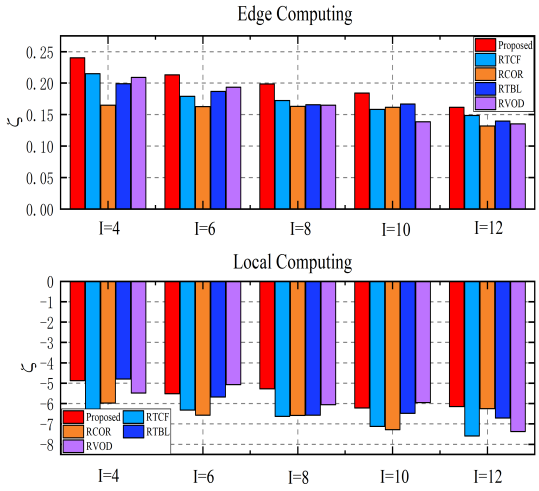

Fig. 8: ζ under different number of vehicles I.

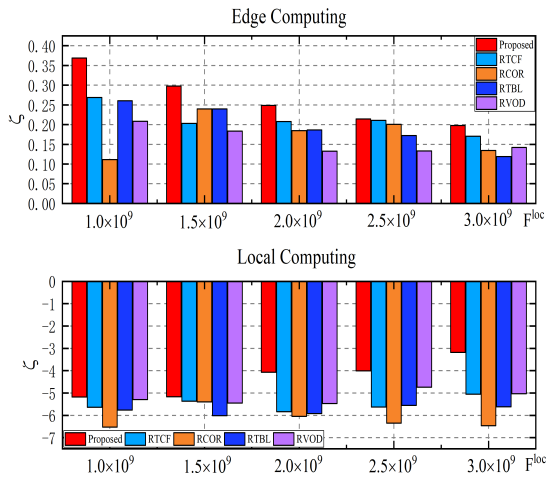

Fig. 9: ζ under different maximum local computation frequency F^{loc}.

possible to fall into the feasible solution until convergence. So, it is possible for Algorithm 3 to converge such quickly as in Fig. 5.

C. Performance of the Proposed Algorithms

As shown in Fig. 6, with the density of attackers in the vehicle set becoming denser, the risk of the offloading transmission process becomes higher, so the secure information capacity of task transmission, which is the optimized objective of offloading computing, will decrease. Similarly, when the number of lanes N_l on the road increases, the possibility of a growing number of attacker vehicles also rises. Therefore, the eight-lane has a lower objective value than the four-lane and the six-lane, and the secure information capacity is lower.

In Fig. 7, we compare the minimum, the average, and the maximum objective value in the local computing and edge computing of the proposed scheme, PTCF, PCOR, RTBL, and RVOD, which correspond to the worst, the average and the best optimization in vehicle set. In edge computing, the higher $\mathcal{O}_i$ means further secure information capacity, and the higher $\mathcal{O}_i$ means lower local computation delay in local computing. It can be seen that the proposed scheme achieves better performance and has a balance between the best and the worst performance in the vehicle set. The PCOR randomly chooses encryption computation frequency, transmit power, and edge computation frequency, so the performance difference is not obvious in edge computing. Moreover, the selection of the weighted value ϖ_1 also makes the difference not prominent.

We can see that the minimum objective value ζ in the vehicle set $\mathcal{I}$ changes with the different number of vehicles from Fig. 8. As the number of vehicles I increases, ζ will decrease, whether it is local computing or edge computing. This is because vehicles of diverse conditions and performances may be included, and the smallest objective value ζ may achieve a

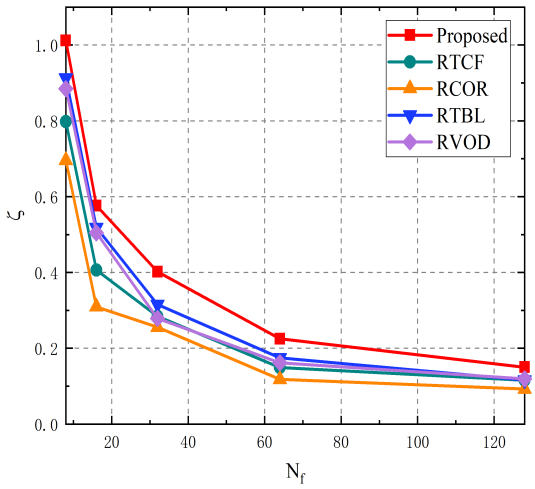

Fig. 10: ζ under different maximum block length N_f.

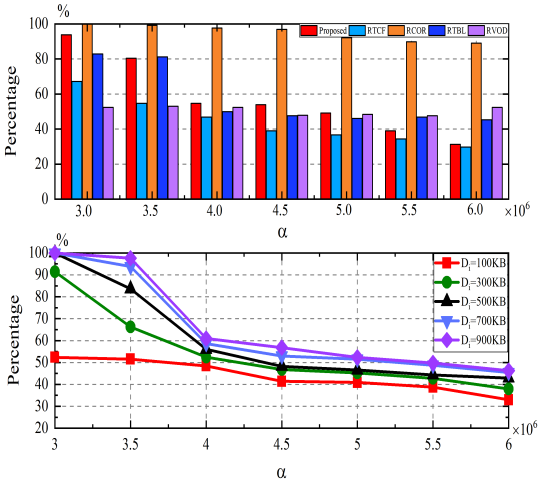

Fig. 11: The number of vehicles selected for local computing as a percentage of all vehicles under different connection state α.

smaller value as the number of vehicles increases. However, the proposed scheme still has better performance than other schemes. It can be seen that the optimization of transmission block length is very necessary, which can greatly improve the performance.

Fig. 9 represents the minimum objective value in the vehicle set $\mathcal{I}$ under different maximum local computation frequency F^{loc}. The proposed scheme obtains the maximum value of ζ in comparison with other schemes. As the maximum local computation frequency becomes larger, the range values of f^c become larger, and its optimal value becomes larger. So ζ becomes larger for local computation. For edge computing, as f^{en} becomes larger, the freedom of taking p_i becomes wider, allowing it to take smaller values. Thus ζ becomes smaller. However, since the PCOR scheme randomly chooses computation offloading resources, such as encryption computation frequency, transmit power, and edge computation frequency, F^{loc} does not have a significant impact on the overall.

Fig. 10 shows the minimum objective value ζ in the vehicle set changes with the different maximum block lengths N_f. When $0 \leq n_i^e < 1$, the optimal block length $N_i{}^*$ depends on the number of attacker vehicles n_i^e, and has nothing to do with N_f. The broader the transmission block length N_f can be, the lower the security strength S_i of the vehicle i, thereby reducing the secure information capacity and the objective value. The PCOR randomly chooses three variables, so the performance is the worst. Although this is the case of edge computing, comparing the optimization of transmission block length and vehicular offloading decision, the optimization benefit of task computation frequency is higher. It influences the process of computation offloading through offloading decisions.

Fig.11 represents the trend of the proportion of vehicles decided to locally compute in the vehicle set as the connection state α_i changes. It can be seen from the two pictures that the vehicles are more willing to offload tasks to the edge server as the connection state gets better. As can be seen from the upper figure of Fig. 11, the PCOR randomly chooses computation offloading resources, such as encryption computation frequency, transmit power, and edge computation frequency, so vehicles decide on local computing. As the connection state becomes better, some vehicles with weak computation capacity will gradually offload tasks to the edge server. The PVOD randomly chooses vehicular offloading decisions, so the statistics are kept at about 50%. The bottom figure in Fig. 11 shows the proportion of vehicles computing locally in the change of different connection states α_i and transmission data size D_i in the proposed scheme. For the same connection state, the larger the data size, the higher the proportion of local computing, because the overall cost of offloading computation is excessive.

V. Conclusions

In this paper, we focused on the joint optimization of security strength and resource allocation for computation offloading in VEC. Taking a full account of the open accessed channels and highly dynamic movement of vehicles, factors such as the driving velocity and position of the vehicles, the number of lanes, the model and density of the attackers, and security strength are added to the system modeling. We established a max-min optimization problem to jointly optimize offloading decision, transmit power, task computation frequency, encryption computation frequency, edge computation frequency, and block length to obtain optimal secure information capacity and local computation delay. The formulated problem is intractable, which is a MINLP. A GBD-based method is applied to solve it. The simulation results showed that our proposed algorithms have well convergence and effectiveness. It is utility to achieve fairness to the secure information capacity and local computation delay among vehicles. Meanwhile, the proposed schemes have a significant performance advantage compared with other schemes.

APPENDIX A
PROOF THE CONVEXITY OF PROBLEM (17).

The variables $\boldsymbol{p}$, $\boldsymbol{f^c}$, $\boldsymbol{f^{en}}$, and $\boldsymbol{f^e}$ in the constraints of the problem (17) are all related with constant by multiples, roots, reciprocals, or exponents, and the linear combination of them. So, the primal problem is the minimum linear objective function with convex constraints satisfying Slater's condition and strong duality.

APPENDIX B
PROOF OF THE SET OF SUBGRADIENTS

Under given Lagrangian dual variables set $\boldsymbol{\Xi} = \{\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\phi}, \varphi, \boldsymbol{\psi}\}$, we can substitute it into (22), (24), (26) and (28) to obtain the optimal variables $\boldsymbol{p}^*$, $\boldsymbol{f^{c*}}$, $\boldsymbol{f^{en*}}$, $\boldsymbol{f^{e*}}$. According the equation (18), we have

$$
\begin{aligned}
&G(\boldsymbol{\Xi}^{'}) \leq f_0 \\
&+\sum_{i=1}^{I}\lambda_i^{'}\Big(x_i\frac{D_iq_i^c}{f_i^{c*}}+(1-x_i)\Big(\frac{D_iq_i^{en}}{f_i^{en*}}+\frac{D_i}{\alpha_i\sqrt{{p_i}^*}}+\frac{D_iq_i^e}{f_i^{e*}}\Big)-\tau_i\Big) \\
&+\sum_{i=1}^{I}\mu_i^{'}\Big(x_ik_iD_iq_i^cf_i^{c*2}+(1-x_i)\Big(k_iD_iq_i^{en}f_i^{en*2}+\frac{D_i\sqrt{{p_i}^*}}{\alpha_i}\Big) \\
&-E_i\Big)+\sum_{i=1}^{I}\beta_i^{'}\Big(f_i^{c*}-F_i^{loc}\Big)+\sum_{i=1}^{I}\phi_i^{'}\Big(f_i^{en*}-F_i^{loc}\Big) \\
&+\varphi^{'}\Big(\sum_{i=1}^{I}f_i^{e*}-F^e\Big)+\sum_{i=1}^{I}\psi_i^{'}\times\Big(\zeta-(1-x_i)\varpi_1\frac{\alpha_i\sqrt{{p_i}^*}}{8N_i} \\
&\times\log_2(8N_i)\Big(\frac{N_i}{N_f}\Big)^{n_i^e}+x_i\varpi_2\frac{D_iq_i^c}{f_i^{c*}}\Big).
\end{aligned}
$$

Adding $G(\boldsymbol{\Xi})$ to both sides of the above inequality, we can obtain the following inequality,

$$
\begin{aligned}
&G(\boldsymbol{\Xi}^{'}) \\
&+\sum_{i=1}^{I}(\lambda_i-\lambda_i^{'})\Big(x_i\frac{D_iq_i^c}{f_i^{c*}}+(1-x_i)\Big(\frac{D_iq_i^{en}}{f_i^{en*}}+\frac{D_i}{\alpha_i\sqrt{{p_i}^*}}+\frac{D_iq_i^e}{f_i^{e*}}\Big) \\
&-\tau_i\Big)+\sum_{i=1}^{I}(\mu_i-\mu_i^{'})\Big(x_ik_iD_iq_i^cf_i^{c*2}+(1-x_i)\Big(k_iD_iq_i^{en}f_i^{en*2} \\
&+\frac{D_i\sqrt{{p_i}^*}}{\alpha_i}\Big)-E_i\Big)+\sum_{i=1}^{I}(\beta_i-\beta_i^{'})\Big(f_i^{c*}-F_i^{loc}\Big) \\
&+\sum_{i=1}^{I}(\phi_i-\phi_i^{'})\Big(f_i^{en*}-F_i^{loc}\Big)+(\varphi-\varphi^{'})\Big(\sum_{i=1}^{I}f_i^{e*}-F^e\Big) \\
&+\sum_{i=1}^{I}(\psi_i-\psi_i^{'})\Big(\zeta-(1-x_i)\varpi_1\frac{\alpha_i\sqrt{{p_i}^*}}{8N_i}\log_2(8N_i)\Big(\frac{N_i}{N_f}\Big)^{n_i^e} \\
&+x_i\varpi_2\frac{D_iq_i^c}{f_i^{c*}}\Big)\leq G(\boldsymbol{\Xi}).
\end{aligned}
$$

According to the definition of the subgradients z of a convex function $g(\cdot)$ at the point x_0: $g(x_0)+z^T(x-x_0)\leq g(x)$, which can be held for all x in the domain, a set of subgradients can be obtained in (31). Therefore, the acquisition of the subgradients set is proved.

APPENDIX C
PROOF THE OPTIMAL SOLUTION OF FUNCTION (38)

By taking the derivative for the function (38), we get the derivative function as follows

$$
\begin{aligned}
&\frac{\partial f(N_i)}{\partial N_i} = \\
&(1-x_i)\varpi_1\frac{\alpha_i\sqrt{p_i}}{8N_f^{n_i^e}}{N_i}^{n_i^e-2}\left(\frac{1}{\ln 2}+(n_i^e-1)\log_2(8N_i)\right).
\end{aligned}
$$

When $n_i^e \geq 1$, $\frac{\partial f(N_i)}{\partial N_i} \geq 0$ always holds, so $f(N_i)$ monotonically increasing. The function can be maximized only when N_i takes the maximum value N_f; When $0 \leq n_i^e<1$ and $\log_2(8N_i) \leq \frac{1}{(1-n_i^e)\ln 2}$, $\frac{\partial f(N_i)}{\partial N_i} \geq 0$ always holds. Also, when $0 \leq n_i^e<1$ and $\log_2(8N_i)>\frac{1}{(1-n_i^e)\ln 2}$, $\frac{\partial f(N_i)}{\partial N_i}<0$ always holds. Hence, $f(N_i)$ increases first and then decreases. The function can be maximized only when $\log_2(8N_i) = \frac{1}{(1-n_i^e)\ln 2}$, i.e., $N_i = 2^{\frac{1}{(1-n_i^e)\ln 2}-3}$. Therefore, the optimal value of N_i can be expressed as follows

$$
{N_i}^* = \begin{cases} 2^{\frac{1}{(1-n_i^e)\ln 2}-3}, & 0 \leq n_i^e<1, \\ N_f, & n_i^e \geq 1. \end{cases}
$$

So, the optimal solution of the function (38) is proved.

REFERENCES

[1] Y. Mansouri and M. A. Babar, "A review of edge computing: Features and resource virtualization," *Journal of Parallel and Distributed Computing*, 2021.

[2] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," *Mobile Networks and Applications*, vol. 26, no. 3, pp. 1145–1168, 2021.

[3] C. Gao, G. Wang, W. Shi, Z. Wang, and Y. Chen, "Autonomous driving security: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7572–7595, 2022.

[4] Y. Cao and Y. Chen, "Qoe-based node selection strategy for edge computing enabled internet-of-vehicles (ec-iov)," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017, pp. 1–4.

[5] J. Feng, Z. Liu, C. Wu, and Y. Ji, "Ave: Autonomous vehicular edge computing framework with aco-based scheduling," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 10 660–10 675, 2017.

[6] R. Wang, F. Zeng, X. Deng, and J. Wu, "Joint computation offloading and resource allocation in vehicular edge computing based on an economic theory: walrasian equilibrium," *Peer-to-Peer Networking and Applications*, pp. 1–13, 2021.

[7] S. Choo, J. Kim, and S. Pack, "Optimal task offloading and resource allocation in software-defined vehicular edge computing," in *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, 2018, pp. 251–256.

[8] Z. Wang, S. Zheng, Q. Ge, and K. Li, "Online offloading scheduling and resource allocation algorithms for vehicular edge computing system," *IEEE Access*, vol. 8, pp. 52 428–52 442, 2020.

[9] A. Pandey, P. Calyam, S. Debroy, S. Wang, and M. L. Alarcon, "Vectrust: Trusted resource allocation in volunteer edge-cloud computing workflows," in *Proceedings of the 14th IEEE/ACM International Conference on Utility and Cloud Computing*, ser. UCC '21. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: https://doi.org/10.1145/3468737.3494099

[10] S. Wang, J. Li, G. Wu, H. Chen, and S. Sun, "Joint optimization of task offloading and resource allocation based on differential privacy in vehicular edge computing," *IEEE Transactions on Computational Social Systems*, pp. 1–11, 2021.

[11] B. Mao, Y. Kawamoto, and N. Kato, "Ai-based joint optimization of qos and security for 6g energy harvesting internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7032–7042, 2020.

[12] Y. Wang, X. Tang, and T. Wang, "A unified qos and security provisioning framework for wiretap cognitive radio networks: A statistical queueing analysis approach," *IEEE Transactions on Wireless Communications*, vol. 18, no. 3, pp. 1548–1565, 2019.

[13] H. Xiao, Q. Pei, X. Song, and W. Shi, "Authentication security level and resource optimization of computation offloading in edge computing systems," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[14] Z. Sun, Y. Liu, J. Wang, W. Deng, and S. Xu, "Non-cooperative game of effective channel capacity and security strength in vehicular networks," *Physical Communication*, vol. 25, pp. 214–227, 2017.

[15] Z. Sun, Y. Liu, J. Wang, R. Yu, and D. Cao, "Cross-layer tradeoff of qos and security in vehicular ad hoc networks: A game theoretical approach," *Computer Networks*, vol. 192, p. 108031, 2021.

[16] M. Shojafar, N. Cordeschi, and E. Baccarelli, "Energy-efficient adaptive resource management for real-time vehicular cloud services," *IEEE Transactions on Cloud Computing*, pp. 196–209, 2016.

[17] S. M. Abuelenin and A. Y. Abul-Magd, "Empirical study of traffic velocity distribution and its effect on vanets connectivity," 2014.

[18] S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Analytical model for connectivity in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3341–3356, 2008.

[19] A. Y., S. Hosny, and A. A. El-Sherif, "Towards mobility-aware proactive caching for vehicular ad hoc networks," 2018.

[20] H. Xiao, J. Zhao, Q. Pei, J. Feng, L. Liu, and W. Shi, "Vehicle selection and resource optimization for federated learning in vehicular edge computing," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, 2021.

[21] M. Shojafar, "Saving energy in qos networked data centers," 2016.

[22] M. Shojafar, N. Cordeschi, D. Amendola, and E. Baccarelli, "Energy-saving adaptive computing and traffic engineering for real-time-service data centers," in *IEEE International Conference on Communication Workshop*, 2015.

[23] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, no. 23, pp. P.2145–2146, 1991.

[24] M. Haleem, C. Mathur, R. Chandramouli, and K. Subbalakshmi, "Opportunistic encryption: A trade-off between security and throughput in wireless networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 4, pp. 313–324, 2007.

[25] J. Thota, N. F. Abdullah, A. Doufexi, and S. Armour, "V2v for vehicular safety applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–15, 2019.

[26] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 3 – 55, 1948.

[27] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *Journal of Vlsi Signal Processing Systems for Signal Image & Video Technology*, vol. 13, no. 2-3, pp. 203–221, 1996.

[28] W. Zhang, Y. Wen, K. Guan, and D. Kilper, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, 2013.

[29] Y. Mao, J. Zhang, S. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5994–6009, 2017.

[30] J. Feng, W. Zhang, Q. Pei, J. Wu, and X. Lin, "Heterogeneous computation and resource allocation for wireless powered federated edge learning systems," *IEEE Transactions on Communications*, vol. 70, no. 5, pp. 3220–3233, 2022.

[31] J. Feng, L. Liu, Q. Pei, and K. Li, "Min-max cost optimization for efficient hierarchical federated learning in wireless edge networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2687–2700, 2022.

[32] H. Xiao, L. Cai, J. Feng, Q. Pei, and W. Shi, "Resource optimization of mab-based reputation management for data trading in vehicular edge computing," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2023.

[33] S. Shah, L. Wang, P. Reddy, and A. Carie, "Non-cooperative game to balance energy and security in resource constrained iot networks," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2020.

Huizi Xiao is currently pursuing a Ph.D. degree in Communication and Information Systems at Xidian University, Xi'an, China. From 2021 to 2023, she was with the University of Victoria, BC, Canada, as a visiting Ph.D. student. Her current research interests include edge computing, resource allocation, security and privacy, convex optimization, and stochastic network optimization.

Jun Zhao (Member, IEEE) received the bachelor's degree from Shanghai Jiao Tong University, China, and the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University (CMU), USA, (advisors: V. Gligor and O. Yagan; collaborator: A. Perrig), affiliating with CMUs renowned CyLab Security and Privacy Institute. Before joining NTU first as a Post-Doctoral Researcher with X. Xiao and then as a Faculty Member, he was a Postdoctoral Researcher with Arizona State University as an Arizona Computing PostDoc Best Practices Fellow (advisors: J. Zhang and V. Poor). He is currently an Assistant Professor with the School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore. His research interests include communications, networks, security, and AI.

Jie Feng (Member, IEEE) received her Ph.D. degree in communication and information systems from Xidian University, China, in 2020. She is currently an Associate Professor at the Department of Electrical Engineering and Computer Science, Xidian University, Xian, China. From 2018 to 2019, she was with Carleton University, Ottawa, ON, Canada, as a visiting Ph.D. student. Her current research interests include mobile-edge computing, blockchain, deep reinforcement learning, the device to device communication, resource allocation and convex optimization, and stochastic network optimization.

Lei Liu (Member, IEEE) received the B.Eng. degree in communication engineering from Zhengzhou University, Zhengzhou, China, in 2010, and the M.Sc. and Ph.D. degrees in communication engineering from Xidian University, Xian, China, in 2013 and 2019, respectively. From 2013 to 2015, he worked in a technology company. From 2018 to 2019, he was supported by China Scholarship Council to be a visiting Ph.D. student with the University of Oslo, Oslo, Norway. He is currently a Lecture with the Department of Electrical Engineering and Computer Science, Xidian University. His research interests include vehicular ad hoc networks, intelligent transportation, mobile-edge computing, and Internet of Things.

Qingqi Pei (Member, IEEE) received his B.S., M.S. and Ph.D. degrees in Computer Science and Cryptography from Xidian University, in 1998, 2005 and 2008, respectively. He is now a Professor and member of the State Key Laboratory of Integrated Services Networks, also a Professional Member of ACM and Senior Member of IEEE, Senior Member of Chinese Institute of Electronics and China Computer Federation. His research interests focus on privacy preserving, blockchain and edge computing security.

Weisong Shi (Fellow, IEEE) is a Professor and Chair of the Department of Computer and Information Sciences at the University of Delaware (UD), where he leads the Connected and Autonomous Research (CAR) Laboratory. Dr. Shi is an internationally renowned expert in edge computing, autonomous driving and connected health. His pioneer paper entitled Edge Computing: Vision and Challenges has been cited more than 5000 times. Before he joined UD, he was a professor at Wayne State University (2002-2022). Dr. Shi has published more than 270 articles in peer-reviewed journals and conferences and served in editorial roles for more than 10 academic journals and publications, including EIC of Smart Health, AEIC of IEEE Internet Computing Magazine. He is a fellow of IEEE, and a distinguished member of ACM. More information can be found at http://weisongshi.org.