Roadside IRS Assisted Task Offloading in Vehicular Edge Computing Network

Yibin Xie[1,2], Lei Shi[1,2(✉)], Zhehao Li[1,2], Xu Ding[1,2], and Feng Liu[1,2]

[1] School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China
shilei@hfut.edu.cn

[2] Engineering Research Center of Safety Critical Industrial Measurement and Control Technology, Ministry of Education, Hefei 230009, China

Abstract. Vehicular edge computing (VEC) has been recognized as a promising technique to process delay-sensitive vehicular applications. Nevertheless, in order to accommodate the rapid growth in the number of connected vehicles, it's inevitable that there will be an increasing deployment of conventional infrastructure with limited communication ranges. This could potentially lead to escalating costs and impede the full realization of the VEC system. In this paper, a roadside intelligent reflecting surface (IRS) assisted VEC network is introduced, where the IRS is deployed outside the coverage of roadside units (RSUs) to extend the service range. Furthermore, the maximum total number of successful offloading tasks problem within the scheduling time problem is formulated, encompassing the optimization of offloading decisions, computation resource allocation and phase shift of IRS. To tackle the formulated challenging problem, we first decouple the original problem into two subproblems. Then, a heuristic algorithm is proposed, where a many-to-one matching algorithm is proposed to joint optimize offloading decision and the computation resource, and an iterative algorithm is utilized to optimize the phase shift coefficients of IRS. The simulation results validate the effectiveness of the proposed algorithm in comparison to other schemes, and the IRS can effectively maintain network performance even when there are intervals in RSU coverage areas.

Keywords: Intelligent Reflecting Surface · Vehicular Edge Computing · Task Offloading · Resource Allocation

1 Introduction

In recent years, with the rapid trend towards intelligence, more vehicular applications such as autonomous driving, traffic prediction and other on-board services are emerging [1,2]. Since most of these applications are with massive data and

This work is supported by major science and technology projects in Anhui Province, NO. 202003a05020009.

H. Gao et al. (Eds.): CollaborateCom 2023, LNICST 561, pp. 365–384, 2024.
https://doi.org/10.1007/978-3-031-54521-4_20

low latency requirements, the computational burden is becoming crazy for vehicles. To cope with the problem, mobile edge computing (MEC) framework has been proposed in vehicular networks, which further inspires the vehicular edge computing (VEC) paradigm [3,4]. The VEC framework allows vehicle to offload and execute their latency-sensitive tasks to the nearby edge servers through vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) or other forms of wireless links, which not only significantly utilizes the computational resources but also improves the quality of service [5–7].

In general, to reap the benefits from VEC, it is crucial to deploy a dedicated access point known as a roadside unit (RSU) in close proximity to users' locations. The RSU can not only exchange information between vehicles and road infrastructure but also enhances vehicular services by integrating edge servers to provide computational support [8]. However, due to the limited communication range of RSU, it is inevitable to deploy a large number of RSUs for fulfilling the service demand from massive connected vehicles [9]. This leads to a subsequent cost concern, covering infrastructure expenses and operating overhead [10]. Moreover, under the fifth-generation (5G) communication background, RSUs employing sophisticated communication techniques tend to incur a lot more energy consumption [11]. Additionally, the rapid growth in the number of connected vehicle users further amplifies this energy burden. Hence, it is necessary to guarantee the communication quality of vehicular networks with low overhead.

Fortunately, owing to the recent progress in programmable meta-materials, a revolutionized technique has emerged, known as the intelligent reflecting surface (IRS), alternatively referred to as the reconfigurable intelligent surface [12]. The IRS is typically composed of a large array of passive reflecting elements, with full control over the phase shift and amplitude of each element [13]. The key function of IRS is to reconfigure wireless channel. More precisely, it can establish a virtual line-of-sight (LoS) link between transceivers to improve communication quality and signal coverage [14]. Since the reflecting elements of IRS are passive, low-cost and easy to deploy, it can enhance the energy and spectral efficiency of wireless networks under little additional cost [15,16]. These benefits have motivated researchers to incorporate IRS into conventional wireless networks, which leads to the emergence of many novel high performance systems, such as IRS-assisted MEC networks and vehicular networks [17,18].

Considering the IRS-assisted MEC scenario, most studies care about the overall offloading capacity and the total energy consumption. The authors in [19] considered an IRS-aided multiuser MEC scenario and propose two algorithms to solve the total energy consumption minimization problem. When combining IRS and other 5G communication technique, Li *et, al.* [20] analyzed sum energy consumption in an IRS-aided multi user MEC system, where users using non-orothology multi access (NOMA). Simulation results show that the proposed RIS-MEC with NOMA scheme can dramatically decrease the sum energy consumption compared to TDMA based methods. Mao *et, al.* [21] proposed an IRS-assisted secure MEC network frame, where eavesdroppers is considered.

Aiming to max-min computation efficiency among all devices, they proposed an iterative algorithm. Simulation results show the proposed method outperforms the conventional scheme without the aid of IRS. The authors in [22] also investigated the secure offloading problem in IRS-assisted MEC network and presented a block coordinate descent algorithm to minimize the total latency. Considering multi base stations scenario, Wang *et, al.* [23] proposed an IRS-assisted edge heterogeneous network, which includes a macro base station and multiple small base station. By introducing a two-timescale mechanism to optimize user association, offloading decision as well as computation resource and phase shift, the long-term energy consumption is minimized.

In terms of the IRS-assisted vehicular networks, existing work mainly focus on deploying IRS for communication link performance improvement. The authors in [24] tried to deploy IRS in a hybrid OMA/NOMA-enabled access environment. By jointly optimizing the transmit power allocation, the IRS's phase shift, and the vehicle active beamforming vector, the total system sum rate is proven to be maximized. Chen *et, al.* in [25] analyzed resource allocation for IRS aided vehicular communications under slowly large-scale fading channel. To maximize V2I link capacity, an alternating optimization algorithm has been proposed. Numerical results demonstrate that the IRS can effectively improve the quality of vehicular communications. For moving vehicular networks, Jiang *et, al.* [26] proposed a novel paradigm coined intelligent reflecting vehicle surface (IRVS) that embeds a massive number of reflection elements on vehicles' surfaces. Simulation results revealed that the IRVS can substantially improve the capacity of moving vehicular networks. The authors in [27] also studied IRS aided mobile vehicles scenario and presented an efficient two-stage transmission protocol for reaping the high IRS passive beamforming gain with low channel training overhead. Simulation results indicate that the proposed method can efficiently achieve the full IRS beamforming gain in the high-mobility communication scenario.

From the perspective of the aforementioned studies, the advantages of IRS are undoubtedly striking. However, the inquiry into whether IRS can meet the requirements of vehicles while being positioned outside the coverage of RSU has not been comprehensively investigated. If the IRS can effectively extend the service range of a RSU, it will have the potential to significantly reduce the costs associated with deploying additional RSUs, particularly in the face of a increasing number of connected vehicles. Motivated by these considerations, we aim to deploy the IRS beyond the coverage of RSU to further explore its ability on expanding service range of VEC network. The main contribution of this article are summarized as follows:

- We propose a roadside IRS-assisted VEC network model, where the IRS is deployed between two RSUs but beyond the coverage of either. An optimization problem is formulated to maximize the number of successful offloading tasks during the scheduling time by jointly optimizing the offloading decision, computation resource allocation and the IRS phase shift coefficients.
- To tackle this challenging problem, we design a heuristic algorithm to first decompose the original problem into two subproblems. Subsequently, the

offloading decision and computation resource allocation are first optimized by solving the first subproblem. Next, the phase shift of IRS is optimized via solving the second subproblem.
– Simulation results show that the proposed algorithm can effectively improve the number of successful offloading compared to other schemes. Besides, the proposed algorithm is able to maintain the network performance even the interval existing between the two RSUs.

The rest of this paper is organized as follows: In Sect. 2, we give the mathematical model of the roadside IRS assisted VEC network and give the corresponding problem formation. In Sect. 3, we design a heuristic algorithm to get a feasible solution. In Sect. 4, we illustrate the simulation results and corresponding analysis. In the final section, we summarize this article.

2 System Model and Problem Formation

In this section, we will give the system model and the problem formation. Consider an intelligent reflecting surface (IRS) assisted vehicular edge computing scenario as shown in Fig. 1, where two single antenna RSUs are deployed with certain distance interval on the side of a unidirectional road with two traffic streams. Suppose each RSU is equipped with a MEC server, so that it can provide proper computing services for vehicle users. We further assume that the coverage radius of both RSUs is smaller than the interval between them, which means when a vehicle is on the road, it may not be covered by any RSUs. So we decide to arrange a IRS to extend the service range of RSU. Suppose the IRS is arranged in the middle of these two RSUs, and it has N reflecting elements which can be controlled by the IRS controller. In the scheduling progress, vehicles would offload their computation tasks to MEC severs through the direct vehicle–RSU links or indirect vehicle–IRS–RSU links. Our main objective is to optimize the total number of successful offloading tasks in the whole scheduling progress. In the following, we will first illustrate the channel model and the offloading model, then give the overall problem model.

2.1 Channel Model

For one vehicle's task, the task completion time includes the uplink transmission time and the execution time. Without loss of generality, we divide the whole schedule time T equally into multiple identical time slots $t(t \in \mathcal{T})$, where $\mathcal{T}$ is the set of time slots. Suppose all channels associated with the mobile vehicles follow block fading, i.e., channels would remain approximately constant during each time slot/block but may vary over different slots/blocks due to the high mobility of vehicles. In addition, we also assume that the channel state information (CSI) of all channels involved is perfectly known by RSUs.

Denote $\mathcal{K}$ as the set of two RSUs, and denote $\mathcal{M}_t$ as the set of vehicles to be served in each slot. For vehicles in the coverage of any RSU, they could

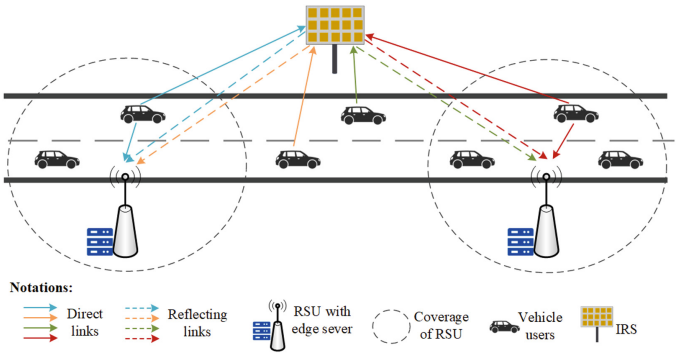

Fig. 1. Overview of the model.

upload their tasks through vehicle–RSU links and vehicle–IRS–RSU links, while for vehicles beyond the coverage of RSUs, they could only upload their tasks via vehicle–IRS–RSU links. Thus, the direct channel from vehicle $m(m \in \mathcal{M}_t)$ to the RSU $k(k \in \mathcal{K})$ in time slot t is denoted as $h_{m,k}(t)$, the channel from vehicle m to the IRS and from IRS to RSU k in time slot t are denoted as $\mathbf{h}_m(t) \in \mathbb{C}^{N\times 1}$ and $\mathbf{g}_k^H(t) \in \mathbb{C}^{1\times N}$, respectively.

Among the three channels described above, the channel $h_{m,k}(t)$ obeys Rayleigh fading and can be expressed as

$$h_{m,k}(t) = \sqrt{\rho(d_{m,k}(t))^{-\lambda}}\, \widetilde{h}_{m,k}(t), \tag{1}$$

where ρ represents the path loss at the reference distance d_0, and λ is the corresponding path loss exponent of vehicle m to RSU k [28]. Furthermore, $d_{m,k}(t)$ represents the distance between vehicle m and RSU k, while $\widetilde{h}_{m,k}(t)$ represents a random scattering component that follows a complex Gaussian distribution with a mean of zero and a variance of one.

Suppose the channel $\mathbf{h}_m(t)$ between IRS and vehicles as well as the channel $\mathbf{g}_k^H(t)$ between RSU and IRS both follow the Rician fading, so $\mathbf{h}_m(t)$ can be modeled as

$$\mathbf{h}_m(t) = \sqrt{\gamma_m(t)}\left(\sqrt{\frac{\xi_m}{1+\xi_m}}\bar{\mathbf{h}}_m(t) + \sqrt{\frac{1}{1+\xi_m}}\widetilde{\mathbf{h}}_m(t)\right), \tag{2}$$

where the vector $\bar{\mathbf{h}}_m(t) \in \mathbb{C}^{N\times 1}$ and vector $\widetilde{\mathbf{h}}_m(t) \in \mathbb{C}^{N\times 1}$ are the line-of-sight (LoS) component and non-line-of-sight (NLoS) component, respectively. To be specific, $\bar{\mathbf{h}}_m(t)$ consists of array response, while $\widetilde{\mathbf{h}}_m(t)$ consists of independent elements that follow $\mathcal{CN}(0,1)$. Besides, ξ_m is the Rician fading factor, which is the ratio of power between the LoS path and the scattered paths, and $\gamma_m(t)$ represents the path loss component from vehicle m to the IRS. It is worth to note that the specific expression of $\mathbf{g}_k(t)$ is similar to (2), which is omitted here due to the limitation of paper length.

The phase shift coefficients of the IRS in time slot t is defined as $\boldsymbol{\theta}(t) = \{\theta_1(t), \ldots, \theta_N(t)\}^T$, and the available value for each phase shift is $[0, 2\pi]$. Then, the reflection matrix of the IRS in time slot t is given by

$$\boldsymbol{\Theta}(t) = \mathrm{diag}(e^{j\theta_1(t)}, e^{j\theta_2(t)}, \cdots, e^{j\theta_N(t)}), \tag{3}$$

where j denotes the imaginary unit. Note that, the amplitudes of all reflecting elements are set to 1. In practice, the phase shifts of the IRS are normally adjusted by the IRS controller using a finite number of discrete values [25]. However, for simplicity of analysis, we assume that the phase shifts can be continuously varied, and we disregard any delays in the adjustment of phase shifts.

2.2 Offloading Model

Suppose that in each time slot, each vehicle may generate one computation task that should be executed on the RSU, and these tasks are delay-sensitive, which means the execution deadline should not greater than the time slot length τ. To determine whether a vehicle is transmitting during slot t and to identify the preferred RSU for offloading, we introduce a binary variable α as

$$\alpha_{m,k}(t) = \begin{cases} 1\ , & \text{if vehicle } m \text{ will offload its task to} \\ & \text{RSU } k \text{ at slot } t; \\ 0\ , & \text{otherwise.} \end{cases} \tag{4}$$

Furthermore, to indicate whether vehicle m is in the coverage of RSU k, we define a binary variable $x_{m,k}(t)$ as

$$x_{m,k}(t) = \begin{cases} 1\ , & \text{if vehicle } m \text{ is in the coverage of} \\ & \text{RSU } k \text{ at slot } t; \\ 0\ , & \text{otherwise.} \end{cases} \tag{5}$$

Assume that vehicles offloading their tasks to the same RSU employ Non-Orthogonal Multiple Access (NOMA) for transmission, which means that they will share the same frequency for communication. In contrast, vehicles offload their tasks to other RSUs will use different frequencies for transmission. This indicates that vehicles with different offloading destinations will not encounter any transmission interference. Thus, we could obtain the uplink signal-plus-interference-noise-ratio (SINR) of vehicle m to RSU k as follows,

$$\Gamma_{m,k}(t) = \frac{p_m \ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_m(t) + h_{m,k}(t)x_{m,k}(t)|^2 \alpha_{m,k}(t)}{\sum\limits_{\substack{m' \in \mathcal{M}_t \\ \pi_{m'} < \pi_m}} p_{m'} \ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_{m'}(t) + h_{m',k}(t)x_{m',k}(t)|^2 \alpha_{m',k}(t) + \sigma^2}, \tag{6}$$

p_m represents the transmit power of vehicle m, and σ^2 denotes the additive white Gaussian noise (AWGN) power. Besides, π_m denotes the successive interference cancellation (SIC) decoding order of signal of vehicle m, and $\pi_{m'} < \pi_m$ means the signal of vehicle-m' will be decoded after vehicle m's signal.

In this way, the transmit rate from vehicle m to RSU k in slot t can be expressed as

$$R_{m,k}(t) = Blog_2(1 + \Gamma_{m,k}(t)), \tag{7}$$

where B is the uplink bandwidth, s_m represents the task bits from vehicle m that need to be offloaded to the RSU. Therefore, the corresponding transmission delay $T_m^{trans}(t)$ is

$$T_m^{trans}(t) = \frac{s_m(t)}{\sum\limits_{k \in \mathcal{K}} R_{m,k}(t)}. \tag{8}$$

We suppose that the two RSUs have different computation capacities but can both implement parallel computing for independent user computations. Therefore, the computation time $T_m^{com}(t)$ to execute vehicle m's task can be expressed as

$$T_m^{com}(t) = \frac{Cs_m(t)}{\sum\limits_{k \in \mathcal{K}} f_{m,k}(t)\ \alpha_{m,k}(t)}, \tag{9}$$

where C is the number of CPU-cycles needed by executing tasks per bit, and $f_{m,k}(t)$ is the computation resource allocated for executing tasks of vehicle m.

Since the size of the computation results are typically small compared to the original tasks, the latency for downloading can be effectively disregarded [29]. Then, the total time delay of completing vehicle m's task is denoted as

$$T_m^{total}(t) = T_m^{trans}(t) + T_m^{com}(t). \tag{10}$$

2.3 Problem Formation

To better evaluate whether the task been has been proceed, we define the following variable μ_m^{total}:

$$\mu_m(t) = \begin{cases} 1\ , & \text{if } T_m^{total}(t) \leq D_m; \\ 0\ , & \text{otherwise,} \end{cases} \tag{11}$$

where D_m denotes the maximum tolerable execution delay of task s_m. Then, the total number of task been executed successfully during the time period T is:

$$\mu^{total} = \sum_{t \in \mathcal{T}} \sum_{m \in \mathcal{M}_t} \mu_m(t). \tag{12}$$

The main objective of this work is to maximize μ^{total} by jointly optimizing the phase shift coefficients (i.e., $\boldsymbol{\theta} = \{\boldsymbol{\theta}(0), \cdots, \boldsymbol{\theta}(t), \cdots\}$), the offloading decision variables (i.e., $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}(0), \cdots, \boldsymbol{\alpha}(t), \cdots\}$), and the computation resource allocation variables (i.e., $\boldsymbol{f} = \{\boldsymbol{f}(0), \cdots, \boldsymbol{f}(t), \cdots\}$). Thus, we have the optimization

problem as follows.

$$\begin{aligned}
\mathcal{P}1: \quad & \max_{\boldsymbol{\alpha},\boldsymbol{\theta},\boldsymbol{f}} \ \mu^{total} \\
& s.t. \quad (1)-(12), \\
& \quad C1: \ \sum_{k\in\mathcal{K}} \alpha_{m,k}(t) \le 1, \ \forall m \in \mathcal{M}_t, \ \forall t \in \mathcal{T}, \\
& \quad C2: \ 0 \le \theta_i(t) \le 2\pi, \ \forall i \in \{1, \cdots, N\}, \ \forall t \in \mathcal{T}, \\
& \quad C3: \ \Gamma_{m,k}(t) > \beta\alpha_{m,k}(t), \ \forall m \in \mathcal{M}_t, \ \forall k \in \mathcal{K}, \ \forall t \in \mathcal{T}, \\
& \quad C4: \ \sum_{m\in\mathcal{M}_t} f_{m,k}(t) \le F_k, \ \forall k \in \mathcal{K}, \ \forall t \in \mathcal{T}, \\
& \quad C5: \ 0 \le f_{m,k}(t) \le F_k, \ \forall k \in \mathcal{K}, \ \forall m \in \mathcal{M}_t, \ \forall t \in \mathcal{T}.
\end{aligned} \tag{13}$$

In (13), constraint $C1$ guarantees that each vehicle's task can only be transmitted to one RSU, while constraint $C2$ restricts the phase shift coefficient for each reflecting elements on IRS. Then, constraint $C3$ gives threshold β of SINR, which makes sure each vehicle could upload their tasks to RSU. Next, constraint $C4$ and $C5$ together ensure proper allocation of computation resource. Specifically, constraint $C4$ ensures that the overall allocated computation resource should not exceed RSU's computation capacity $F_k(k \in \mathcal{K})$, while constraint $C5$ guarantees the allocated resource for each vehicle's task.

According to (6), (7), (8), (9), the binary variables $\boldsymbol{\alpha}$ and the continuous variables $\boldsymbol{\theta}$ and $\boldsymbol{f}$ are deeply coupled with each other. Therefore, problem $\mathcal{P}1$ is a mix-integer non-convex problem, which is NP-hard and challenging to solve. In the following, we will propose a heuristic algorithm to solve this problem.

3 The Proposed Solution

In this section, we will introduce a two-stage heuristic algorithm to tackle the problem $\mathcal{P}1$. To begin with, it is important to notice that the variables $\boldsymbol{\alpha}(t)$, $\boldsymbol{\theta}(t)$, $\boldsymbol{f}(t)$ in each time slot is independent. Therefore, to get the maximum value of μ^{total}, we only need to maximize $\sum_{m\in\mathcal{M}_t} \mu_m(t)$ in each time slot. In this way, we can obtain the following optimization problem in each time slot:

$$\begin{aligned}
\mathcal{P}2: \quad & \max_{\boldsymbol{\alpha}(t),\boldsymbol{\theta}(t),\boldsymbol{f}(t)} \ \sum_{m\in\mathcal{M}_t} \mu_m(t) \\
& s.t. \quad (1)-(11), \\
& \quad C1-C5.
\end{aligned} \tag{14}$$

Problem $\mathcal{P}2$ has the same feature as problem $\mathcal{P}1$, which is non-convex and hard to solve. Therefore, we first divide the problem $\mathcal{P}2$ into two subproblems: 1) the optimization problem $\mathcal{P}3$ of offloading decision $\boldsymbol{\alpha}(t)$ and computation resource allocation $\boldsymbol{f}(t)$; 2) the optimization problem $\mathcal{P}4$ of phase shift coefficients $\boldsymbol{\theta}(t)$. To solve problem $\mathcal{P}2$, we propose a many-to-one matching method under fixed phase shift vector $\boldsymbol{\theta}(t)$. For solving $\mathcal{P}3$, we propose a iterative phase shift optimization method by fixing the variable $\boldsymbol{\alpha}(t)$ and $\boldsymbol{f}(t)$. The proposed optimization algorithm for IRS assisted VEC network is summarized in Algorithm 1. In the following, we will discuss in detail.

Algorithm 1. The Overall Algorithm

Input: $\mathcal{M}_t$: The set of vehicles in slot t, $\mathcal{K}$: The set of RSUs, $\mathcal{T}$: The set of time slots.
Output: $\boldsymbol{\alpha}^*$, $\boldsymbol{f}^*$, $\boldsymbol{\theta}^*$.
1: **for** $t \in \mathcal{T}$ **do**
2: Initialize phase shift coefficients $\boldsymbol{\theta}^0(t)$.
3: Obtain offloading decision $\boldsymbol{\alpha}^*(t)$ and resource allocation $\boldsymbol{f}^*(t)$ based on $\boldsymbol{\theta}^0(t)$ via **Algorithm 2**.
4: Find feasible phase shift $\boldsymbol{\theta}^*(t)$ based on $\boldsymbol{\alpha}^*(t)$ and $\boldsymbol{f}^*(t)$ via **Algorithm 3**.
5: Compute $\mu_m^*(t)$ according to $\boldsymbol{\alpha}^*(t)$, $\boldsymbol{f}^*(t)$, $\boldsymbol{\theta}^*(t)$.
6: Let $\mu^*(t) = \sum_{m \in \mathcal{M}_t} \mu_m^*(t)$.
7: **end for**

3.1 Optimization of Offloading Decisions and Computation Resource Allocation

When fixing the IRS phase shift coefficients $\boldsymbol{\theta}(t)$, problem $\mathcal{P}2$ can be reduced as

$$\begin{aligned} \mathcal{P}3: \quad & \max_{\boldsymbol{\alpha}(t),\ \boldsymbol{f}(t)} \sum_{m \in \mathcal{M}_t} \mu_m(t), \\ & s.t. \quad (1) - (11), \\ & \quad C1, C3, C4, C5. \end{aligned} \tag{15}$$

Note that, the problem $\mathcal{P}3$ remains non-convex and hard to be solved directly. However, according to (11) the value of $\mu_m(t)$ is related to the total offloading latency $T_m^{total}(t)$ of each vehicle. If each $T_m^{total}(t)$ keeps a relatively small value, the number of successful offloading tasks will increase. In this way, problem $\mathcal{P}3$ can be further transferred as

$$\begin{aligned} \tilde{\mathcal{P}}3: \quad & \min_{\boldsymbol{\alpha}(t),\ \boldsymbol{f}(t)} \sum_{m \in \mathcal{M}_t} T_m^{total}(t), \\ & s.t. \quad (1) - (11), \\ & \quad C1, C3, C4, C5. \end{aligned} \tag{16}$$

Now, we focus on finding the optimal $\boldsymbol{\alpha}(t)$ and $\boldsymbol{f}(t)$ to minimize computation time for each vehicle. However, according to (6), (7), (8), (9), it is evident to find variable $\boldsymbol{\alpha}(t)$ and $\boldsymbol{f}(t)$ are deeply coupled, which makes problem $\tilde{\mathcal{P}}3$ is still computationally hard to cope with.

According to constraint $C1$, we know that each vehicle can only select one RSU as its offloading destination, while a RSU can be assigned to multiple vehicles. Therefore, we decide to utilize the many-to-one matching in each time slot for getting the optimal $\boldsymbol{\alpha}(t)$ and $\boldsymbol{f}(t)$.

Definition 1. Suppose η is a many-to-one matching function that joins vehicle set $\mathcal{M}_t$ and RSU set $\mathcal{K}$. The function η should satisfy the following conditions:

1. $|\eta(m)| = 1$ for every vehicle $m \in \mathcal{M}_t$;

2. $|\eta(k)| \leq |\mathcal{M}_t|$ for every RSU $k \in \mathcal{K}$;
3. $m \in \eta(k)$ if and only if $k = \eta(m)$.

In fact, the matching η can represent the offloading decision $\boldsymbol{\alpha}(t)$ in the considered scenario. Then, based on **Definition 1.**, we define $U_m(\eta)$ and $U_k(\eta)$ as the utility function for m on η and the utility function for k on η. The two utility functions actually reveal the matching effect. We have

$$U_m(\eta) = -T_m^{\eta}, \tag{17}$$

where T_m^{η} is the total executing latency of vehicle m as shown in (10) under the matching η. Besides, $U_k(\eta)$ is expressed as

$$U_k(\eta) = -\max_{m \in \mathcal{M}_t} \{T_m^{\eta}\}. \tag{18}$$

The above utility functions (17) and (18) show both vehicles and RSUs care about more than their own matching. So, traditional pairwise stable matching may not exist in such condition [30]. To obtain the desire stable matching, we further leverage the concept of two-sided swap stability on the following definitions.

Definition 2. Define $\eta_m^{m'} = \{\eta \backslash \{(m,k),(m',k')\}\} \cup \{(m,k'),(m',k)\}$ as a swap matching, where $\eta(m) = k$, $\eta(m') = k'$, and $k \neq k'$.

Normally, a swap matching allows vehicle m and vehicle m' to swap their corresponding matched RSUs with each other, while remaining the matchings between other vehicles and RSUs stable. However, when $k' = 0$, the swap matching will be modified as $\eta_m^0 = \{\eta \setminus (m,k)\} \cup \{(m,k')\}$, where $\eta(m) = k$ and $k \neq k'$. This means that for vehicle m, the originally matched RSU k is now changed to k'.

Definition 3. Given a matching function η and a pair of vehicles (m, m'), if there exists $\eta(m) = k$ and $\eta(m') = k'$, and satisfies:

1. $\forall z \in \{m, m', k, k'\}, U_z(\eta_{m,}^{m'}) \geq U_z(\eta)$;
2. $\exists z \in \{m, m', k, k'\}, U_z(\eta_{m}^{m'}) \geq U_z(\eta)$;

then we call (m, m') as a swap-blocking pairs in matching η.

Definition 4. A matching is reckoned to be two-sided exchange stable if and only if there is no swap-blocking pairs exists in η.

Definition 4 is a criterion that evaluates whether the current matching is stable and guarantees the convergence of many-to-one matching method.

Nevertheless, to proceed the many-to-one steps, we need to tackle the computation resource allocation problem. After obtaining any matching during the whole many-to-one matching process, the computation resource allocation problem $\mathcal{P}3.1$ can be expressed as follows:

$$\begin{aligned} \mathcal{P}3.1: \quad & \min_{\boldsymbol{f}(t)} \sum_{m \in \mathcal{M}_t} T_m^{com}(t), \\ & s.t. \quad (9),\ C4, C5. \end{aligned} \tag{19}$$

Algorithm 2. The ODRA Algorithm

Input: $\mathcal{M}_t$: The set of vehicles in slot t, $\mathcal{K}$: The set of RSUs, $\boldsymbol{\theta}^0(t)$: The initial set of phase shift coefficients.
Output: $\boldsymbol{\alpha}^*(t)$, $\boldsymbol{f}^*(t)$.
Initialize a matching η on $\mathcal{M}_t \cup \mathcal{K}$.
Initialize computation resource allocation $\boldsymbol{f}(t)$ based on initial matching η.
repeat
  $\exists m \in \mathcal{M}_t,\ \eta(m) = k$.
  Vehicle m choose another RSU k' and re-solve problem $\tilde{\mathcal{P}}3.1$ to obtain $\boldsymbol{f}'(t)$.
  Compute corresponding SINR $\Gamma_{m,k'}(t)$ according to (6).
  if vehicle m satisfies $U_m(\eta_m^0) \geq U_m(\eta)$ and $\Gamma_{m,k'}(t) \geq \beta$ **then**
    if $U_k(\eta_m^0) \geq U_k(\eta)$ or $U_{k'}(\eta_m^0) \geq U_{k'}(\eta)$ **then**
      Update matching: $\eta \leftarrow \eta_m^0$.
      Update resource allocation: $\boldsymbol{f}(t) = \boldsymbol{f}'(t)$.
    else
      Do not update η.
      The computation resource allocation $\boldsymbol{f}(t)$ remains.
    end if
  end if
  $\exists m, m' \in \mathcal{M}_t$, and $m \neq m', \eta(m) = k, \eta(m') = k'$.
  Vehicle m and vehicle m' swapping matched RSUs and re-solve problem $\tilde{\mathcal{P}}3.1$ to obtain $\boldsymbol{f}'(t)$.
  Compute corresponding SINR $\Gamma_{m,k'}(t)$ and $\Gamma_{m',k}(t)$ according to (6).
  if vehicle m and m' satisfy $U_m(\eta_m^{m'}) \geq U_m(\eta)$ and $U_{m'}(\eta_m^{m'}) \geq U_{m'}(\eta)$ and $\Gamma_{m,k'}(t) \geq \beta$ and $\Gamma_{m',k}(t) \geq \beta$ **then**
    if $U_k(\eta_m^{m'}) \geq U_k(\eta)$ or $U_{k'}(\eta_m^{m'}) \geq U_{k'}(\eta)$ **then**
      Update matching: $\eta \leftarrow \eta_m^{m'}$.
      Update offloading decision: $\boldsymbol{f}(t) = \boldsymbol{f}'(t)$.
    else
      Do not update η.
      The computation resource allocation $\boldsymbol{f}(t)$ remains.
    end if
  end if
until Matching η satisfies **Definition 4**.
Let $\boldsymbol{\alpha}^*(t) = \eta$, $\boldsymbol{f}^*(t) = \boldsymbol{f}(t)$.

The above problem is non-convex due to the objective function, thus we further transfer problem $\mathcal{P}3.1$ into:

$$\tilde{\mathcal{P}}3.1: \quad \max_{\boldsymbol{f}(t)} \sum_{m \in \mathcal{M}_t} \frac{1}{T_m^{com}(t)}, \quad s.t. \quad (9),\ C4, C5. \tag{20}$$

Now, $\tilde{\mathcal{P}}3.1$ is a convex optimization problem, which can be solved directly through existing solving tools like CVX. Based on the above analysis, the proposed offloading decision and computation resource allocation (ODRA) algorithm is summarized in Algorithm 2.

3.2 Optimization of IRS Phase Shift Coefficients

After obtaining the optimized offloading decision $\boldsymbol{\alpha}^*(t)$ and computation resource allocation $\boldsymbol{f}^*(t)$, we now focus on optimizing the phase shift of IRS. It is evident that we can obtain the total computation latency $\sum_{m\in\mathcal{M}_t} T_m^{com}(t)$ after executing Algorithm 2. So, in order to get the max value of $\sum_{m\in\mathcal{M}_t} \mu_m(t)$, we only need to minimize the total transmission latency $\sum_{m\in\mathcal{M}_t} T_m^{trans}(t)$. Furthermore, minimizing $\sum_{m\in\mathcal{M}_t} T_m^{trans}(t)$ is approximated as maximizing the overall transmission rate of each vehicle. Therefore, the original problem $P2$ can be transferred to

$$\begin{aligned} \mathcal{P}4: \quad & \max_{\boldsymbol{\theta}(t)} \sum_{m\in\mathcal{M}_t}\sum_{k\in\mathcal{K}} R_{m,k}(t) \\ & s.t. \quad (1)-(8),\ C2, C3. \end{aligned} \tag{21}$$

Since the objective function of $\mathcal{P}4$ is non-concave, we try to introduce the slack variables $\boldsymbol{V} = \{V_{m,k}(t) = \frac{p_m\ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_m(t)+h_{m,k}(t)x_{m,k}(t)|^2\alpha_{m,k}(t)}{\sum_{\substack{m'\in\mathcal{M}_t \\ \pi_{m'}<\pi_m}} p_{m'}\ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_{m'}(t)+h_{m',k}(t)x_{m',k}(t)|^2\alpha_{m',k}(t)+\sigma^2}, \forall m,k,t\}$ and then transform it into a feasibility-check problem and solve it iteratively until converge. And the variables $\boldsymbol{V}$ satisfy that

$$\frac{p_m\ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_m(t)+h_{m,k}(t)x_{m,k}(t)|^2\alpha_{m,k}(t)}{\sum_{\substack{m'\in\mathcal{M}_t \\ \pi_{m'}<\pi_m}} p_{m'}\ |\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_{m'}(t)+h_{m',k}(t)x_{m',k}(t)|^2\alpha_{m',k}(t)+\sigma^2} \geq V_{m,k}(t), \quad \forall m\in\mathcal{M}_t, k\in\mathcal{K}, t\in\mathcal{T}. \tag{22}$$

Thus, problem $\mathcal{P}4$ can be reformulated to

$$\begin{aligned} \bar{\mathcal{P}}4: \quad & \text{Find} \quad \boldsymbol{\Phi}(t) \\ & s.t. \quad (1)-(8), (22)\ C2, C3. \end{aligned} \tag{23}$$

Due to the non-convexity of constraint $C3$ and (22), we will do the following transformation. The combined channel gain in (6) of each vehicle can be reformulated as

$$|\mathbf{g}_k^H(t)\boldsymbol{\Theta}(t)\mathbf{h}_m(t)+h_{m,k}(t)x_{m,k}(t)|^2 = |\mathbf{H}_m(t)\boldsymbol{\Phi}(t)+h_{m,k}(t)x_{m,k}(t)|^2, \tag{24}$$

where $\mathbf{H}_m(t) = \mathbf{g}_k^H(t)\text{diag}\{\mathbf{h}_m(t)\}$ and $\boldsymbol{\Phi}(t) = [e^{j\theta_1(t)}, e^{j\theta_2(t)}, \cdots, e^{j\theta_N(t)}]^T$. We then introduce two slack variables $\kappa_m(t)$ and $\zeta_m(t)$, which are defined as [31]:

$$\kappa_m(t) = \mathrm{Re}(\mathbf{H}_m(t)\boldsymbol{\Phi}(t) + h_{m,k}(t)x_{m,k}(t)),\ \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T}, \tag{25}$$

$$\zeta_m(t) = \mathrm{Im}(\mathbf{H}_m(t)\boldsymbol{\Phi}(t) + h_{m,k}(t)x_{m,k}(t)),\ \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T}. \tag{26}$$

The variables $\kappa_m(t)$ and $\zeta_m(t)$ stand for the real part and imaginary part, respectively, and they satisfy the relationship $\kappa_m(t)^2 + \zeta_m(t)^2 = |\mathbf{H}_m(t)\boldsymbol{\Phi}(t) + h_{m,k}(t)x_{m,k}(t)|^2$. To this end, problem $\bar{\mathcal{P}}4$ can be rewritten as below

$$\begin{aligned}
\tilde{\mathcal{P}}4: \quad & \text{Find} \quad \boldsymbol{\Phi}(t) \qquad (27)\\
& s.t. \quad (1)-(8),(24)-(26),\\
& C6: |\Phi_i(t)| \le 1,\ \forall i \in \{1,\cdots,N\}, \forall t \in \mathcal{T},\\
& C7: \frac{p(\kappa_m(t)^2+\zeta_m(t)^2)\alpha_{m,k}(t)}{\sum\limits_{\substack{m' \in \mathcal{M}_t \\ \pi_{m'} < \pi_m}} p(\kappa_{m'}(t)^2+\zeta_{m'}(t)^2)\alpha_{m',k}(t)+\sigma^2} \ge \beta, \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T},\\
& C8: \frac{p(\kappa_m(t)^2+\zeta_m(t)^2)\alpha_{m,k}(t)}{\sum\limits_{\substack{m' \in \mathcal{M}_t \\ \pi_{m'} < \pi_m}} p(\kappa_{m'}(t)^2+\zeta_{m'}(t)^2)\alpha_{m',k}(t)+\sigma^2} \ge V_{m,k}(t),\\
& \quad \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T}.
\end{aligned}$$

$\Phi_i(t)$ represents the ith component in $\boldsymbol{\Phi}(t)$.

Note that, problem $\tilde{\mathcal{P}}4$ is still non-convex due to the constraint $C7$ and $C8$. So, we decide to adopt the successive convex approximation method to deal with the non-convexity [32]. At given initial point $(\tilde{\kappa}_m(t),\ \tilde{\zeta}_m(t))$, according to the first-order Taylor expansion of $\kappa_m(t)^2 + \zeta_m(t)^2$, we can get

$$\begin{aligned}
\kappa_m(t)^2 + \zeta_m(t)^2 \ge \tilde{\kappa}_m(t)^2 + \tilde{\zeta}_m(t)^2 + 2\tilde{\kappa}_m(t)(\kappa_m(t) - \tilde{\kappa}_m(t))\\
+2\tilde{\zeta}_m(t)(\zeta_m(t) - \tilde{\zeta}_m(t)) = \Gamma_m^{low}(t).
\end{aligned} \tag{28}$$

Therefore, problem $\tilde{\mathcal{P}}4$ can be further formulated as

$$\begin{aligned}
\hat{\mathcal{P}}4: \quad & \text{Find} \quad \boldsymbol{\Phi}(t) \qquad (29)\\
& s.t. \quad (1)-(8),(24)-(26),(28),\\
& C6: |\Phi_i(t)| \le 1,\ \forall i \in \{1,\cdots,N\}, \forall t \in \mathcal{T},\\
& C9: \frac{p\Gamma_m^{low}(t)\alpha_{m,k}(t)}{\sum\limits_{\substack{m' \in \mathcal{M}_t \\ \pi_{m'} < \pi_m}} p(\kappa_{m'}(t)^2+\zeta_{m'}(t)^2)\alpha_{m',k}(t)+\sigma^2} \ge \beta, \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T},\\
& C10: \frac{p\Gamma_m^{low}(t)\alpha_{m,k}(t)}{\sum\limits_{\substack{m' \in \mathcal{M}_t \\ \pi_{m'} < \pi_m}} p(\kappa_{m'}(t)^2+\zeta_{m'}(t)^2)\alpha_{m',k}(t)+\sigma^2} \ge V_{m,k}(t),\\
& \quad \forall m \in \mathcal{M}_t, \forall t \in \mathcal{T}.
\end{aligned}$$

Problem $\hat{\mathcal{P}}4$ is a convex optimization problem, and can be solved effectively through CVX. The detailed steps involving in phase shifts optimization (PSO) algorithm is summarized in Algorithm 3.

Algorithm 3. The PSO Algorithm

Input: $\mathcal{M}_t$: The set of vehicles in slot t, $\mathcal{K}$: The set of RSUs, $\boldsymbol{\theta}^0(t)$: Initial phase shift coefficient in slot t, $\boldsymbol{\alpha}^*(t)$: Optimal offloading decision in slot t, $\boldsymbol{f}^*(t)$: Optimal computation resource allocation in slot t.
Output: $\boldsymbol{\theta}(t)^*$.
1: Set $r = 0$.
2: Compute $\boldsymbol{\Phi}^0(t)$ according to $\boldsymbol{\theta}^0(t)$.
3: **repeat**
4: $r = r + 1$.
5: Update $\tilde{\zeta}_m(t)$ and $\tilde{\kappa}_m(t)$ based on $\boldsymbol{\Phi}^{r-1}(t)$ according to (25) & (26).
6: Solve problem $\hat{\mathcal{P}}4$ to obtain $\boldsymbol{\Phi}^r(t)$, $\zeta_m^r(t)$, $\kappa_m^r(t)$.
7: **until** $\boldsymbol{\Phi}^r(t)$, $\zeta_m^r(t)$, $\kappa_m^r(t)$ coverage.
8: Let $\boldsymbol{\theta}^*(t) = \boldsymbol{\Phi}^r(t)^T$.

4 Numerical Results

In this section, simulation results are presented to verify the effectiveness of the proposed optimization algorithm. To begin with, we consider a two-way lane road where RSU and IRS are arranged on opposite sides. Under a three-dimensional Cartesian coordinate system, the IRS is loacted at (0, 0, 20), while two RSUs with 200m coverage radius locate at (20, –300, 20) and (20, 300, 20), respectively. So, the interval between two RSUs is 200 m. To generate vehicles, we use the traffic flow dataset "MIDAS Site - 19542 at A2260 southbound between A2 and B262", which is from a freeway in Untied Kingdom. And the vehicles' coordinates are in the region between two RSUs, with the y-axis coordinate from –400 to 400. The x-axis coordinate is 0 to 5 and the z-axis coordinate is fixed as 0. Then, the path loss at the reference distance $d_0 = 1$m is configured as –20 dB, and the Rician factor is set as 3 dB. The path loss exponents for vehicle–RSU link, RSU–IRS link and vehicle–IRS link are set as 3.5, 2.2 and 2.2 [29], respectively. Other simulation parameters are shown in Table 1.

In addition, the length of each time slot is set as 1 s, and the overall number of time slots is 10. The task is reckoned as succeed only when the total delay of the task is less than or equal to the maximum tolerable delay, otherwise the task executed fails. All the experimental results are averaged from 50 groups of corresponding experiments.

In the simulations, the following three schemes are involved into the comparison:

- **Without IRS:** In such method, the IRS will not be placed in the considered VEC network, so the total successful number of tasks will only be maximized through optimizing offloading decision and resource allocation
- **With PSO without ODRA:** The ODRA algorithm that relates to offloading decision and resource allocation are not included in such method, but the phase shifts of IRS will be optimized.

Table 1. Simulation parameters

Symbol	Simulation parameter	Value
s_m	Task data amount for each vehicle	$[4 \times 10^5,\ 6 \times 10^5]$ bits
D_m	Delay constraint of each task	[0.5, 1] s
F	The computing capability of two edge servers	15 GHz, 20 GHz
C	Number of CPU-cycles needed by executing tasks per bit	800 cycles/bit
p_m	The transmission power of each vehicle	1 W
B	Uplink bandwidth of each vehicle	10 MHz
β	The threshold for SINR	0.1
σ^2	The Gaussian white noise power	–100 dBm

– **With ODRA without PSO:** The phase shifts of IRS will not be optimized in such scheme, but the ODRA algorithm will be adopted to optimize offloading decision and computation resource.

We first present detailed results of our algorithm under different number of total tasks during the 10 time slots, and compare it with other three schemes. Then, we show the effect of the number of reflection elements on the result of all methods. At last, we indicate the relationship between the location of IRS and the number of total successful offloading tasks.

In Fig. 2, the number of successful offloading tasks versus the various number of total tasks during the 10 time slots is illustrated. These results are presented considering the number of reflecting elements $N = 30$ and the interval between RSUs is 200 m. Notably, all three schemes that incorporate IRS exhibit substantial performance advantages compared to the scheme without IRS, and our proposed algorithm is always the most superior. This outcome effectively showcases the capability of deploying IRS to expand the service coverage of RSUs, as well as the efficacy of our proposed algorithm. Moreover, when the number of generated tasks during 10 slots is below 300, the outcome for all schemes increases with the number of generated tasks rises. In such condition, the three schemes adopting IRS have the similar results. However, when the number of tasks surpasses 300, the results for all schemes consistently grow, except for the "without IRS" scheme. This is because the "without IRS" scheme has a longer average transmission time than other methods. As the number of tasks increases, the average computation time also rises, leading to an increasing number of tasks exceeding their time constraints. Besides, the "with ODRA without PSO" scheme has slightly better performance than the "with PSO without ODRA" scheme, which indicates the importance of correct offloading decisions.

In the second simulation, we investigate the impact of the number of reflection elements while keeping the number of generated tasks fixed. The total number of generated tasks is set at 500, and the RSU interval is maintained at 200m.

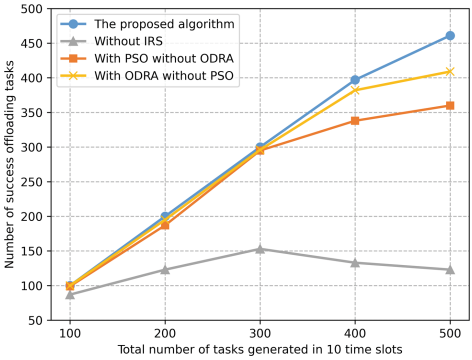

Fig. 2. The number of successful offloading tasks versus the number of tasks generated in 10 time slots, interval = 200 m, $N = 30$.

Subsequently, we set the number of reflecting elements N ranging from 2 to 10 and then from 10 to 50. As shown in Fig. 3 (a), it can be observed that the number of successful offloading tasks under three IRS-assisted schemes increase with the augmentation of N and eventually remains unchanged. Notably, the proposed algorithm and the "with PSO without ODRA" algorithm achieves converge with $N = 6$, while the "with ODRA without PSO" algorithm achieves converge until $N = 8$. This outcome indicates the effectiveness of phase shift optimization method, which helps to obtain the best performance with relatively little number of reflecting elements. Similar to Fig. 3 (a), the situation in Fig. 3 (b) also prove the completion above, as the two schemes employing phase shifts optimization get the maximum performance quicker. However, when the number of reflecting elements exceeds 50, the proposed algorithm and the "with PSO without ODRA" algorithm rises their performance again. In addition, combining both figures, the results for all three IRS aided schemes first remain stable from $N = 8$ and $N = 10$, then rising again. This phenomenon is attributed to a significant reduction in transmission latency when $N = 20$.

In the final simulation experiment, we explore the influence of intervals between RSUs. In this simulation, we keep the reflecting element number $N = 30$ and the total number of generated tasks as 500. To alter the interval, we adjust the y-axis coordinate of the two RSUs from 200 to 400 and –200 to –400, respectively. As shown in Fig. 4, one can observe that as the interval between two RSUs increase, the amount of successful offloading tasks corresponding decreases. Nevertheless, the proposed algorithm remains its maximum performance unchanged until interval surpasses 200 m. In contrast, neither the other two IRS-assisted schemes nor the without IRS scheme manage to achieve similar effects. Furthermore, the performance of without IRS scheme equals our proposed algorithm at the beginning, but experiences a significant drop when the interval is 100 m, being outperformed by other schemes. This simulation result demonstrates that the efficacy of our proposed algorithm in enhancing communication quality and expanding the service range of RSUs. Notably, the "with ODRA without

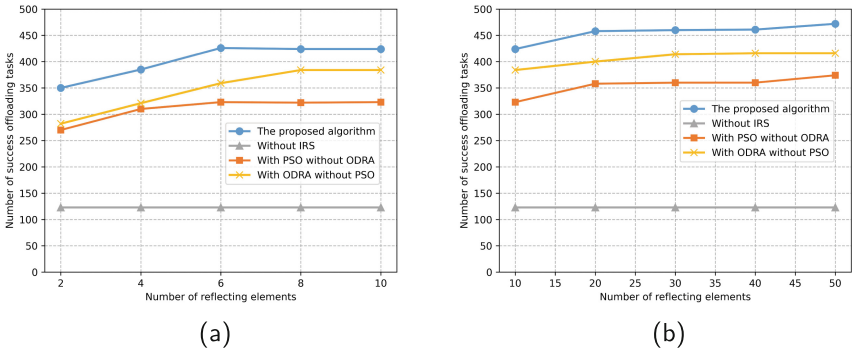

Fig. 3. The number of successful offloading tasks versus the number of reflecting elements, interval = 200 m.

PSO" always outperforms the "with PSO without ODRA" algorithm until the interval between two RSUs reach 400 m. This outcome demonstrates that the phase shifts optimization is more important with the increasing interval.

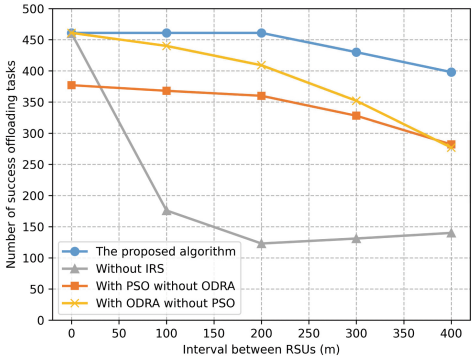

Fig. 4. The number of successful offloading tasks versus the interval between RSUs, N = 30.

5 Conclusion

In this paper, a roadside IRS aided VEC network is proposed, where the IRS is positioned between two RSUs but outside the coverage area of either. Moreover, in order to explore IRS's potential ability on expanding the service range in the considered model, we aim to maximize the number of tasks completed during

a certain period. Since the formulated problem is challenging to cope with, we first decouple the original problem into two subproblems and then propose a heuristic algorithm to jointly optimize the offloading strategy, the computation resource allocation and the phase shifts of IRS. Simulation results show that the proposed algorithm can effectively maximize the overall success offloading tasks, especially compared to the scheme without IRS. It is also notable that the proposed algorithm can maintain its performance even there is interval between two RSUs. Since the positive impact of IRS in extending the RSU's service range has been proved, we will further investigate its potential benefits of enhancing energy efficiency. Additionally, cooperation between multiple IRSs and partial vehicle offloading will be considered.

References

1. Munawar, S., Ali, Z., Waqas, M., Tu, S., Hassan, S.A., Abbas, G.: Cooperative computational offloading in mobile edge computing for vehicles: A model-based DNN approach. IEEE Trans. Veh. Technol. **72**(3), 3376–3391 (2023). https://doi.org/10.1109/TVT.2022.3217323
2. Slamnik-Kriještorac, N., Peeters, M., Latré, S., Marquez-Barja, J.M.: Analyzing the impact of vim systems over the mec management and orchestration in vehicular communications. In: 2020 29th International Conference on Computer Communications and Networks (ICCCN), pp. 1–6 (2020). https://doi.org/10.1109/ICCCN49398.2020.9209636
3. Meneguette, R., De Grande, R., Ueyama, J., Filho, G.P.R., Madeira, E.: Vehicular edge computing: Architecture, resource management, security, and challenges. ACM Comput. Surv. **55**(1) (2021). https://doi.org/10.1145/3485129
4. Liu, L., Chen, C., Pei, Q., Maharjan, S., Zhang, Y.: Vehicular edge computing and networking: a survey. Mob. Netw. Appl. **26**, 1145–1168 (2019). https://api.semanticscholar.org/CorpusID:201070569
5. Zhang, J., Guo, H., Liu, J., Zhang, Y.: Task offloading in vehicular edge computing networks: a load-balancing solution. IEEE Trans. Veh. Technol. **69**(2), 2092–2104 (2020). https://doi.org/10.1109/TVT.2019.2959410
6. Lin, H., Zeadally, S., Chen, Z., Labiod, H., Wang, L.: A survey on computation offloading modeling for edge computing. J. Netw. Comput. Appl. **169**, 102781 (2020). https://doi.org/10.1016/j.jnca.2020.102781, https://www.sciencedirect.com/science/article/pii/S1084804520302551
7. Jaiswal, N., Purohit, N.: Performance analysis of NOMA-enabled vehicular communication systems with transmit antenna selection over double nakagami-m fading. IEEE Trans. Veh. Technol. **70**(12), 12725–12741 (2021). https://doi.org/10.1109/TVT.2021.3119979
8. Geng, L., Zhao, H., Wang, J., Kaushik, A., Yuan, S., Feng, W.: Deep-reinforcement-learning-based distributed computation offloading in vehicular edge computing networks. IEEE Internet Things J. **10**(14), 12416–12433 (2023). https://doi.org/10.1109/JIOT.2023.3247013
9. Yu, H., Liu, R., Li, Z., Ren, Y., Jiang, H.: An RSU deployment strategy based on traffic demand in vehicular ad hoc networks (VANETs). IEEE Internet Things J. **9**(9), 6496–6505 (2022). https://doi.org/10.1109/JIOT.2021.3111048

10. Ghosh, S., Misra, I.S., Chakraborty, T.: Optimal RSU deployment using complex network analysis for traffic prediction in VANET. Peer-to-Peer Network. Appl. **16**, 1135–1154 (2023). https://api.semanticscholar.org/CorpusID:257435229
11. Leung, V.C.M., Dong, Y., Pan, H.: Editorial recent techniques of green information and communications technologies. IEEE Trans. Green Commun. Netw. **5**(4), 1649–1652 (2021). https://doi.org/10.1109/TGCN.2021.3125232
12. Chen, R., Liu, M., Hui, Y., Cheng, N., Li, J.: Reconfigurable intelligent surfaces for 6g IoT wireless positioning: a contemporary survey. IEEE Internet Things J. **9**(23), 23570–23582 (2022). https://doi.org/10.1109/JIOT.2022.3203890
13. Liao, Y., Xia, S., Zhang, K., Zhai, X.: UAV swarm trajectory and cooperative beamforming design in double-IRS assisted wireless communications. In: 2022 18th International Conference on Mobility, Sensing and Networking (MSN), pp. 594–600 (2022). https://doi.org/10.1109/MSN57253.2022.00099
14. Wu, Q., Zhang, S., Zheng, B., You, C., Zhang, R.: Intelligent reflecting surface-aided wireless communications: a tutorial. IEEE Trans. Commun. **69**(5), 3313–3351 (2021). https://doi.org/10.1109/TCOMM.2021.3051897
15. Gong, S., Lu, X., Hoang, D.T., Niyato, D., Shu, L., Kim, D.I., Liang, Y.C.: Toward smart wireless communications via intelligent reflecting surfaces: a contemporary survey. IEEE Commun. Surv. Tutorials **22**(4), 2283–2314 (2020). https://doi.org/10.1109/COMST.2020.3004197
16. Zheng, B., You, C., Mei, W., Zhang, R.: A survey on channel estimation and practical passive beamforming design for intelligent reflecting surface aided wireless communications. IEEE Commun. Surv. Tutorials **24**(2), 1035–1071 (2022). https://doi.org/10.1109/COMST.2022.3155305
17. Chen, N., Liu, C., Jia, H., Okada, M.: Intelligent reflecting surface aided network under interference toward 6g applications. IEEE Network **36**(4), 18–27 (2022). https://doi.org/10.1109/MNET.001.2100675
18. Cao, Y., Xu, S., Liu, J., Kato, N.: Toward smart and secure v2x communication in 5g and beyond: a UAV-enabled aerial intelligent reflecting surface solution. IEEE Veh. Technol. Mag. **17**(1), 66–73 (2022). https://doi.org/10.1109/MVT.2021.3136832
19. Chu, Z., Xiao, P., Shojafar, M., Mi, D., Mao, J., Hao, W.: Intelligent reflecting surface assisted mobile edge computing for internet of things. IEEE Wirel. Commun. Lett. **10**(3), 619–623 (2021). https://doi.org/10.1109/LWC.2020.3040607
20. Li, Z., et al.: Energy efficient reconfigurable intelligent surface enabled mobile edge computing networks with NOMA. IEEE Trans. Cogn. Commun. Netw. **7**(2), 427–440 (2021). https://doi.org/10.1109/TCCN.2021.3068750
21. Mao, S., et al.: Reconfigurable intelligent surface-assisted secure mobile edge computing networks. IEEE Trans. Veh. Technol. **71**, 6647–6660 (2022). https://api.semanticscholar.org/CorpusID:247689318
22. Chen, X., Xu, H., Zhang, G., Chen, Y., Li, R.: Secure computation offloading assisted by intelligent reflection surface for mobile edge computing network. Phys. Commun. **57**, 102003 (2023). https://doi.org/10.1016/j.phycom.2023.102003, https://www.sciencedirect.com/science/article/pii/S187449072300006X
23. Wang, Z., Wei, Y., Feng, Z., Yu, F., Han, Z.: Resource management and reflection optimization for intelligent reflecting surface assisted multi-access edge computing using deep reinforcement learning. IEEE Trans. Wirel. Commun. **22**, 1175–1186 (2023). https://api.semanticscholar.org/CorpusID:252963387
24. Salem, A.A., Rihan, M., Huang, L., Benaya, A.: Intelligent reflecting surface assisted hybrid access vehicular communication: Noma or OMA contributes the

most? IEEE Internet Things J. **9**(19), 18854–18866 (2022). https://doi.org/10.1109/JIOT.2022.3162787
25. Chen, Y., Wang, Y., Zhang, J., Li, Z.: Resource allocation for intelligent reflecting surface aided vehicular communications. IEEE Trans. Veh. Technol. **69**(10), 12321–12326 (2020). https://doi.org/10.1109/TVT.2020.3010252
26. Jiang, W., Schotten, H.D.: Intelligent reflecting vehicle surface: a novel IRS paradigm for moving vehicular networks. In: MILCOM 2022–2022 IEEE Military Communications Conference (MILCOM), pp. 793–798 (2022). https://doi.org/10.1109/MILCOM55135.2022.10017691
27. Huang, Z., Zheng, B., Zhang, R.: Transforming fading channel from fast to slow: intelligent refracting surface aided high-mobility communication. IEEE Trans. Wireless Commun. **21**(7), 4989–5003 (2022). https://doi.org/10.1109/TWC.2021.3135685
28. Li, S., Duo, B., Yuan, X., Liang, Y.C., Di Renzo, M.: Reconfigurable intelligent surface assisted UAV communication: Joint trajectory design and passive beamforming. IEEE Wirel. Commun. Lett. **9**(5), 716–720 (2020). https://doi.org/10.1109/LWC.2020.2966705
29. Bai, T., Pan, C., Deng, Y., Elkashlan, M., Nallanathan, A., Hanzo, L.: Latency minimization for intelligent reflecting surface aided mobile edge computing. IEEE J. Sel. Areas Commun. **38**(11), 2666–2682 (2020). https://doi.org/10.1109/JSAC.2020.3007035
30. Liu, Z., Wang, K., Zhou, M.T., Shao, Z., Yang, Y.: Distributed task scheduling in heterogeneous fog networks: a matching with externalities method. In: 2020 International Conference on Computing, Networking and Communications (ICNC), pp. 620–625 (2020). https://doi.org/10.1109/ICNC47757.2020.9049775
31. Yang, Y., Zheng, B., Zhang, S., Zhang, R.: Intelligent reflecting surface meets OFDM: protocol design and rate maximization. IEEE Trans. Commun. **68**(7), 4522–4535 (2020). https://doi.org/10.1109/TCOMM.2020.2981458
32. Razaviyayn, M.: Successive convex approximation: analysis and applications (2014). https://api.semanticscholar.org/CorpusID:59834031