Low-latency Data Computation of Inland Waterway USVs for RIS-Assisted UAV MEC Network

Yangzhe Liao, Yuanyan Song, Shuang Xia, Yi Han, Ning Xu, Xiaojun Zhai and Zhenhui Yuan

***Abstract*—Unmanned Surface Vehicles (USVs) in inland waterways have drawn increasing attention for their excellent capability to serve maritime time-consuming missions such as autonomous navigation and intelligent monitoring. However, USVs struggle to accomplish emerging computation-intensive tasks (e.g., sensor, telemetry, etc) timely due to the limited on-board resources. This paper proposes a novel reconfigurable intelligent surface (RIS)-assisted unmanned aerial vehicle (UAV) multi-access edge computing (MEC) network architecture to support low-latency USVs data computation with time window. Aiming to enhance USVs task processing efficiency, the minimization of USVs task processing time is formulated by jointly considering UAVs flight route selection, USVs execution mode selection, UAVs hovering coordinates and RIS phase shift vector. A heuristic solution is proposed to tackle the formulated challenging problem iteratively. The original problem is decoupled into three subproblems: an enhanced deferred acceptance algorithm is proposed to solve UAVs flight route selection subproblem; an enhanced Lagrangian relaxation method is proposed to solve USVs execution mode selection subproblem; a joint alternating direction method of multipliers (ADMM)-successive convex approximation (SCA)-based algorithm is proposed to solve UAVs hovering coordinates subproblem. Experiment results demonstrate that the proposed solution can decrease task processing time by approximately 54% compared with numerous selected advanced algorithms. Moreover, the performance of the proposed solution under typical UAVs caching capability and the number of UAVs has been investigated.**

***Index Terms*—Unmanned Surface Vehicles, Unmanned Aerial Vehicles, Multi-access Edge Computing, Reconfigurable Intelligent Surface, Time Window**

I. Introduction

A. Research Background and Motivation

Recently, unmanned surface vehicles (USVs) have been gaining significant attention in the maritime industry, including autonomous navigation, environment monitoring, search-and-rescue and so forth [1-2]. On the one hand, USVs are expected to integrate with a variety of advanced sensors, such as light detection and ranging, inertial navigation systems, which are of great significance to realizing full functions in wireless waterway communications. On the other hand, multi-access edge computing (MEC) can significantly reduce USVs energy consumption since data offloading enable resource-hungry USVs to transfer computation tasks to MEC servers, which are connected with terrestrial base stations (TBSs) via optical fiber. As a result, USVs operational hours can be remarkably prolonged [3-4]. Currently, USVs suffer the following major technical challenges: (1) with the emergence of bidirectional computation tasks, such as autonomous navigation, multimedia transmission, intelligent monitoring, path planning and so on, USVs are battery empowered and suffer limited on-board resources regarding computing and communication capabilities to support massive amounts of data process [5-6]. (2) wireless data transmission between USVs and TBS experiences remarkable attenuation, reflections, diffractions and scattering due to specific features of inland river environments. Such characteristics may result in degraded communication performance; some latency-sensitive offloaded tasks may fail since they cannot be transmitted to MEC servers in time [7-8].

RIS-assisted unmanned aerial vehicle (UAV) MEC networks have attracted excessive attention from both academia and industry to support USVs low-latency data computation [9]. Owing to the wide deployment of tethered unmanned aerial vehicles (UAVs) in inland waterway environments, UAVs can achieve stable power supply with prolonged lifetime and perform as flying MEC servers to serve USVs reliably. Moreover, this emerging technology is not only able to enhance USVs computing capability but also to improve their task execution efficiency, especially in decreasing bidirectional computation task processing time cost. However, the design of RIS-assisted UAV MEC networks experiences numerous distinctive challenges, which are elaborated as follows. First, phase shift vector of each UAV-mounted RIS should be appropriately designed, which is challenging to accomplish since RIS passive reflections need to be jointly considered with UAVs flight route selection indicator to enhance their data transmission over the reconfigured wireless links. Moreover, the commonly used USVs execution selection solutions for RIS-assisted UAV MEC networks cannot directly apply to wireless inland waterway communications since they are primarily counted on alternating-based optimization methods, suffering sophisticated algorithm design and extremely high time complexity [10-11]. In a nutshell, how to decrease USVs bidirectional data computation time cost in RIS-assisted UAV MEC networks is still in an early stage and is worth further investigating.

This work was supported in part by the Natural Science Foundation of China under Grant No. 52201417; in part by National Key R&D Program of China under Grant No. 2021ZD0114600; and in part by Shenzhen Science and Technology Program under Grant No. JCYJ20220818102002005.

Y. Liao, Y. Song, S. Xia, Y. Han and N. Xu are with School of Information Engineering, Wuhan University of Technology, Wuhan, China, 430070.

X. Zhai is with School of Computer Science and Electronic Engineering, University of Essex, Colchester, UK, CO4 3SQ.

Z. Yuan is with School of Engineering, University of Warwick, Coventry, UK, CV4 7AL.

(Corresponding author: Yi Han (email: hanyi@whut.edu.cn))

B. Related Works

1) RIS-assisted UAV MEC Networks: One should be aware that traditional MEC networks suffer significant technical challenges regarding communication and computation cooperation since MEC servers are deployed with fixed coordinates that cannot provision on-demand service promptly. Fortunately, UAVs have received substantial attention as supplementary to traditional MEC networks thanks to their excellent flexibility and easy deployment [12]. Aiming to overcome the potential conflicts between QoS requirements and limited UAV onboard resources, one applicable way is to collaborate MEC technology and tethered UAVs to boost network performance. The authors of [13] proposed an MEC-enabled UAV network architecture, where each UAV can perform as an MEC server or a relay based on computing demands. The authors of [14] proposed that low-altitude MEC-empowered UAVs can not only exploit their mobility but also be able to execute latency-sensitive tasks timely. The operation of MEC-empowered tethered UAVs in conjunction with RIS has gained significant attention as one capable and affordable method for fulfilling ubiquitous wireless communication services in rural places. The authors of [15] proved that the network computing capability could be significantly enhanced by integrating RIS and UAVs in the proposed RIS-enabled UAV-assisted MEC network. The authors of [3] proposed a single-cell RIS-assisted wireless inland ship MEC network architecture, where each MEC-empowered UAV is integrated with RIS. In this way, the number of successfully executed tasks could be remarkably enhanced since latency-sensitive tasks can be executed by UAV while computation-intensive tasks can be forwarded via UAV-mounted RIS. The authors of [16] presented an RIS-assisted UAV network architecture and jointly optimized UAVs trajectory and RIS phase shifts using a deep reinforcement learning method. Aiming to enhance the downlink achievable data rate, the authors of [17] proposed that the design of UAV trajectory and RIS phase shift vector could be jointly considered.

2) Look Forward of Inland Waterway USVs in RIS-assisted UAV MEC Networks: The computation model in most existing works on RIS-assisted UAV MEC networks is based on one-way computation task model. The authors of [18] proposed an efficient dependent task offloading algorithm for MEC network by determining the priorities of tasks according to their predetermined completion times. The authors of [19] investigated task offloading scheduling with strict deadlines in MEC, which proved to be NP-hard and extremely challenging to tackle. Aiming to provision time-constrained computation services in UAV-empowered MEC networks, the authors of [20] proposed a novel resource allocation approach by jointly optimizing UAV trajectory and computation offloading. However, the inefficiency of the network performance caused by wireless links between TBS, UAV, and USV has not been thoroughly studied. Due to the harsh inland waterway environments, the direct links between USVs and TBS are easily disturbed or obstructed, which results in high offloading time cost of USVs. Moreover, even though deploying RIS-mounted UAVs can bring flexible communication and computing services to USVs, the performance enhancement of USVs task processing is still challenging since UAVs flight route design and RIS phase shift vector are closely coupled with USVs execution selection. The situation may become more complicated with the rapid developments of USVs data traffic, the bidirectional computation task has recently emerged as a vital use case of USVs, where the input data of each bidirectional computation task consists of two parts, one of which is generated from USVs sensors and the rest originates from the Internet proactively. As such, the design of RIS-assisted UAV MEC networks considering USVs bidirectional task model may become even more challenging. The authors of [21] proposed a novel homogeneous bidirectional task model in MEC. However, this system only considered one MEC server and one mobile device. A more realistic heterogeneous bidirectional computation task model was proposed in [22], where tasks are of different data sizes and computation loads. The results showed that the proposed solution could directly extend to one MEC server and multiple devices scenario. The authors of [5] proposed a novel solution by jointly optimizing RIS phase shift vector, USVs task execution mode selection, UAVs trajectory and task execution latency constraints. However, this solution only considered bidirectional computation tasks with soft time window constraints. In general, USVs bidirectional computation tasks with target hard window for RIS-assisted UAV MEC networks are of great significance to realizing full functions in wireless inland waterway communication; however, very few thorough studies have been focused on this emerging problem due to the extreme difficulty.

C. The Main Contributions and Organizations

According to the above-mentioned research background, unlike the majority of the research works that assume UAVs can only be implemented in one execution mode, e.g., UAV execution mode or UAV-mounted RIS mode with soft task execution time window, this paper investigates low-latency data computation of inland waterway battery-empowered USVs for RIS-assisted UAV MEC network, where each tethered UAV is equipped with an MEC server and RIS. In this way, each tethered UAV can perform as a flying MEC server or assist data transmission via UAV-mounted RIS. Moreover, this paper considers USVs bidirectional task model with hard time window. The main contributions of this paper are summarized as follows.

- This paper proposes a novel RIS-assisted UAV MEC network architecture. In this system, each tethered UAV is connected with TBS via cable to receive a substantial energy source and provision heavy load capacity to carry RIS. Moreover, each USV can dynamically select task execution mode according to the task data size and communication link quality. UAV-mounted RIS is capable of assisting reliable data transmission between TBS and USVs without installing new telecommunications infrastructure.
- The minimization of USVs task processing time is formulated by jointly considering USVs execution selection, RIS phase shift vector, UAVs flight route selection indicator and UAVs hovering coordinates. A heuristic solution is proposed

to solve the formulated problem. In particular, the original optimization problem is divided into three subproblems, where UAVs flight route selection indicator subproblem is solved via the proposed enhanced deferred acceptance (EDA) algorithm; after obtaining the feasible UAVs flight route selection indicator, USVs execution selection subproblem is divided into a series of parallel subproblems, which can be efficiently solved by using the proposed enhanced Lagrangian relaxation method (ELRM); an alternating direction method of multipliers (ADMM)-successive convex approximations (SCA)-based algorithm is proposed to optimize UAVs hovering coordinates and RIS phase shift vector. In this way, the challenging formulated problem can be efficiently solved iteratively.

The rest of the paper is organized as follows. The network architecture and the formulated optimization problem are given in Section II. The proposed solution is presented in Section III. The performance evaluation of the proposed solution along with numerous selected advanced algorithms is shown in Section IV. Section V concludes the paper and gives some promising future research directions.

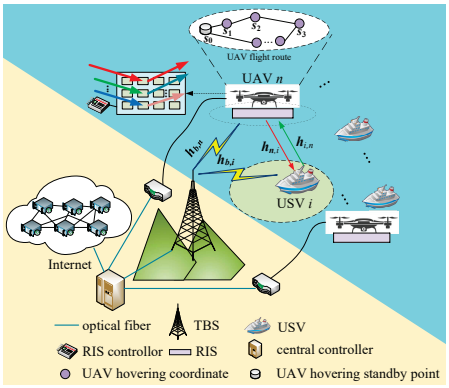

Fig. 1: The proposed network architecture.

II. System Model

The proposed RIS-assisted UAV MEC network architecture is shown in Fig. 1, where one TBS is equipped with L antennas and each rotary-wing single-antenna tethered UAV $n \in \mathcal{N}$ is equipped with K RIS reflecting elements. Assume during each equal-length time slot, each battery-empowered USV $i \in \mathcal{M}$ requests a computation task U_i to be executed, which can be characterized by a 5-tuple: $U_i \triangleq \{D_i^l, D_i^o, F_i, [X_i, Y_i]\}$, where D_i^l and D_i^o indicate the input data size (in bits) generated by USV i and originated from the Internet, respectively. F_i denotes the required number of CPU cycles to execute U_i. $[X_i, Y_i]$ specifies the time window constraints of U_i, where X_i and Y_i denote the earliest starting time and the deadline to execute U_i, respectively. Note that tethered UAVs can obtain reliable power supply from TBS via cable and offer prolonged operational lifetime.

In this paper, the system is established based on a 3D Cartesian coordinate, where the coordinates of TBS and each USV i can be denoted by $\boldsymbol{q_b} \in \mathbb{R}^3$ and $\boldsymbol{q_i} \in \mathbb{R}^3$, respectively. In accordance with [4], each UAV is assumed to fly at a fixed altitude H with a time-varying horizontal coordinate. Moreover, each USV i has a corresponding hovering coordinate $\boldsymbol{s_i} \in \mathbb{R}^3$ when served by UAV [23]. The long-term statistical channel state information of UAV-mounted RIS transmission links can be obtained using the location information aided method [24], where each UAV-USV, TBS-UAV and TBS-USV link are assumed as controllable non-line-of-sight (NLOS), line-of-sight (LOS) and NLOS, respectively. In addition, MEC server can compute RIS phase shift vector and UAVs hovering coordinates according to the channel condition and computing dynamics. Note that each computation task is indivisible and can only be executed in one place. Moreover, TBS is connected to the Internet via optical fiber and thus the data downloading latency is neglected. Table I lists the selected significant notations and descriptions used in this paper.

TABLE I: The selected significant notations used in this paper.

Notation	Definition
$\boldsymbol{\theta_{n,i}}$	RIS phase shift vector of UAV n to serve USV i
$\boldsymbol{\Theta_{n,i}}$	reflection coefficient of UAV n to serve USV i
$\boldsymbol{h_{b,i}}$	channel gain from TBS to USV i
$\boldsymbol{h_{b,n}}$	channel gain from TBS to UAV n
$\boldsymbol{h_{n,i}}$	channel gain from UAV n to USV i
$\boldsymbol{h_{i,n}}$	channel gain from USV i to UAV n
$B_{b,n,i}$	TBS allocated bandwidth of USV i
$B_{i,n}$	UAV allocated bandwidth of USV i
$C_{b,n,i}$	instantaneous RIS-assisted downlink channel capacity
$C_{i,n}$	instantaneous channel capacity between USV i and UAV n
$d_{b,n}$	distance between TBS and UAV n
$d_{n,i}$	distance between UAV n and USV i
$d_{b,n}^{max}$	maximum distance between TBS and UAV n
$d_{n,i}^{max}$	maximum distance between UAV n and USV i
$\beta_n^{i,j}$	flight route selection between $\boldsymbol{s_i}$ and $\boldsymbol{s_j}$ of UAV n
$\tau_{n,i}$	arriving time of UAV n to arrive $\boldsymbol{s_i}$
D_n^{max}	maximum caching capability of UAV n
$\mathcal{M}_n$	service set of UAV n

In this system, each UAV is capable of delivering the following two task execution modes:

Local execution mode: When each USV i decides to execute U_i by itself, one can utilize UAV-mounted RIS to assist data transmission between TBS and each USV i. After receiving the corresponding online data D_i^o, each USV i can start to execute U_i.

UAV execution mode: UAVs perform as flying MEC servers and each USV i offloads U_i to UAV for task execution. In particular, each UAV caches the corresponding online data to serve USVs before taking off from UAV standby point. Then, UAV can start to execute each U_i after receiving the offloaded data from each USV i.

A. Bidirectional Data Computation Model

Let α_i be the task execution selection indicator of USV i, where $\alpha_i = 0$ and $\alpha_i = 1$ indicate that U_i is executed via UAV execution mode and local execution mode, respectively. One has

$$\mathcal{C}1 : \alpha_i \in \{0, 1\}, i \in \mathcal{M}. \tag{1}$$

Local execution mode: When USV i decides to execute U_i by itself, USV i needs to download D_i^o from TBS via UAV-mounted RIS. Assume UAV n is assigned to serve USV i, the corresponding RIS phase shift vector to serve USV i can

be denoted by $\boldsymbol{\theta_{n,i}} = [\theta_{n,i}^1, \theta_{n,i}^2, \ldots, \theta_{n,i}^K]^T$, where each RIS element k of $\boldsymbol{\theta_{n,i}}$ should follow

$$\mathcal{C}2 : \theta_{n,i}^k \in [0, 2\pi], k \in \{1, 2, \ldots, K\}, i \in \mathcal{M}, n \in \mathcal{N}. \quad (2)$$

In accordance with [6], we assume that each RIS element follows the full reflection. The reflection coefficient matrix of RIS can be expressed as

$$\boldsymbol{\Theta_{n,i}} = \begin{bmatrix} e^{j\theta_{n,i}^1} & 0 & \cdots & 0 \\ 0 & e^{j\theta_{n,i}^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{j\theta_{n,i}^K} \end{bmatrix}. \quad (3)$$

The baseband channels, TBS $\rightarrow$ USV i, UAV-mounted RIS $\rightarrow$ USV i and TBS $\rightarrow$ UAV-mounted RIS are denoted by $\boldsymbol{h_{b,i}} \in \mathbb{R}^{L\times 1}$, $\boldsymbol{h_{n,i}} \in \mathbb{R}^{K\times 1}$, $\boldsymbol{h_{b,n}} \in \mathbb{R}^{L\times K}$, respectively. $\boldsymbol{h_{b,i}} \in \mathbb{R}^{L\times 1}$ can be expressed as

$$\boldsymbol{h_{b,i}} = \sqrt{\rho_0 d_{b,i}^{-3.5}}\boldsymbol{I}, i \in \mathcal{M}, \quad (4)$$

where ρ_0 represents the channel power at the reference distance. $\boldsymbol{I}$ is the identity vector. $d_{b,i}$ is the distance between TBS and USV i, which can be expressed as $d_{b,i} = ||\boldsymbol{q_i} - \boldsymbol{q_b}||$. $\boldsymbol{h_{n,i}} \in \mathbb{R}^{K\times 1}$ can be expressed as

$$\begin{aligned} \boldsymbol{h_{n,i}} =& \sqrt{\rho_0 d_{n,i}^{-2}}[1, e^{-j\frac{2\pi d}{\lambda}\phi_{n,i}}, \ldots, e^{-j\frac{2(K-1)\pi d}{\lambda}\phi_{n,i}}], \\ & i \in \mathcal{M}, n \in \mathcal{N}, \end{aligned} \quad (5)$$

where $\phi_{n,i}$ is the cosine of the incident angle between UAV-mounted RIS to USV i. λ is the carrier wavelength and d indicates the distance between two successive RIS reflecting elements. Let $d_{n,i}$ be the distance between UAV n and USV i, which can be expressed as $d_{n,i} = ||\boldsymbol{s_i} - \boldsymbol{q_i}||$. $\boldsymbol{h_{b,n}} \in \mathbb{R}^{L\times K}$ can be expressed as[1]

$$\boldsymbol{h_{b,n}} = \sqrt{\rho_0 d_{b,n}^{-2}} \begin{bmatrix} 1 & e^{-j\frac{2\pi d}{\lambda}\phi_{1,n}} & \cdots & e^{-j\frac{2(K-1)\pi d}{\lambda}\phi_{1,n}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{2\pi d}{\lambda}\phi_{l,n}} & \cdots & e^{-j\frac{2(K-1)\pi d}{\lambda}\phi_{l,n}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{2\pi d}{\lambda}\phi_{L,n}} & \cdots & e^{-j\frac{2(K-1)\pi d}{\lambda}\phi_{L,n}} \end{bmatrix}, \quad (6)$$

where each $\phi_{l,n}$ indicates the cosine of the incident angle from l-th antenna of TBS to UAV-mounted RIS. Let $d_{b,n}$ be the distance between TBS and UAV n, which can be expressed as $d_{b,n} = ||\boldsymbol{s_i} - \boldsymbol{q_b}||$.

The received signal $y_i \in \mathbb{C}^{L\times 1}$ by USV i can be expressed as

$$y_i = \sqrt{p_b^{tr}}(\boldsymbol{h_{b,i}} + \boldsymbol{h_{b,n}}\boldsymbol{\Theta_{n,i}}\boldsymbol{h_{n,i}})s_i + \sigma, i \in \mathcal{M}, n \in \mathcal{N}, \quad (7)$$

where p_b^{tr} is the transmission power of TBS and s_i is the transmitted data symbol with average unity power, i.e., $\mathbb{E}(|s_i|^2) = 1$. σ denotes the noise.

[1]According to Snell's law, the incident angle can be reasonably assumed equal to the reflected angle when each RIS element follows the full reflection.

The channels between different USVs are assumed as orthogonal, where the corresponding SNR of each USV i can be expressed as

$$\gamma_i = \frac{p_b^{tr}||\boldsymbol{w_i^H}(\boldsymbol{h_{b,i}} + \boldsymbol{h_{b,n}}\boldsymbol{\Theta_{n,i}}\boldsymbol{h_{n,i}})||^2}{\sigma^2||\boldsymbol{w_i^H}||^2}, i \in \mathcal{M}, n \in \mathcal{N}, \quad (8)$$

where $\boldsymbol{w_i} \in \mathbb{R}^{L\times 1}$ indicates the downlink beamforming vector designed by TBS to serve each USV i. $\boldsymbol{w_i^H}$ is the Hermitian matrix of $\boldsymbol{w_i}$. The instantaneous RIS-assisted downlink channel capacity can be given as

$$C_{b,n,i} = B_{b,n,i}\log_2(1 + \gamma_i), i \in \mathcal{M}, n \in \mathcal{N}, \quad (9)$$

where $B_{b,n,i}$ is the allocated bandwidth to serve USV i. The corresponding time cost to transmit D_i^o can be expressed as

$$t_{b,n,i} = \frac{D_i^o}{C_{b,n,i}}, i \in \mathcal{M}, n \in \mathcal{N}. \quad (10)$$

The local execution time cost of USV i to execute U_i can be given as

$$t_i^l = \frac{F_i}{f_i}, i \in \mathcal{M}, \quad (11)$$

where f_i denotes the computation capability of USV i. Note that each USV i can only start to execute U_i after receive D_i^o.

UAV execution mode: When USV i selects to execute U_i via UAV execution mode, UAV n caches D_i^o when taking off from UAV standby point and serving each USV i at its corresponding hovering coordinate $\boldsymbol{s_i}$. The offloading time cost of USV i to offload D_i^l to UAV n can be expressed as

$$t_{i,n} = \frac{D_i^l}{C_{i,n}}, i \in \mathcal{M}, n \in \mathcal{N}, \quad (12)$$

where $C_{i,n}$ denotes the instantaneous channel capacity between USV i and UAV n, which can be expressed as

$$C_{i,n} = B_{i,n}\log_2(1 + \frac{p_i^{tr}h_{i,n}}{d_{n,i}^2\sigma^2}), i \in \mathcal{M}, n \in \mathcal{N}, \quad (13)$$

where $B_{i,n}$ is the allocated bandwidth of USV i served by UAV n and $h_{i,n}$ denotes the channel gain between each USV i and UAV n. p_i^{tr} indicates the transmission power of USV i.

The corresponding execution time to execute U_i can be given as

$$t_i^n = \frac{F_i}{f_n}, i \in \mathcal{M}, n \in \mathcal{N}, \quad (14)$$

where f_n denotes the computing capability of UAV n.

B. UAV Operation Model

To maintain the channel quality, the distance between each USV i and UAV n cannot exceed the maximum available communication distance $d_{n,i}^{max}$. One has

$$\mathcal{C}3 : ||\boldsymbol{s_i} - \boldsymbol{q_i}|| \leq d_{n,i}^{max}, i \in \mathcal{M}, n \in \mathcal{N}. \quad (15)$$

The distance between TBS and UAV n cannot exceed the maximum available communication distance $d_{b,n}^{max}$. One has

$$\mathcal{C}4 : ||\boldsymbol{s_i} - \boldsymbol{q_b}|| \leq d_{b,n}^{max}, i \in \mathcal{M}, n \in \mathcal{N}. \quad (16)$$

Let $\beta_n^{i,j}$ be flight route selection indicator of each UAV n, where $\beta_n^{i,j} = 1$ represents UAV n selects to fly from any two

successive hovering coordinates, e.g., $\boldsymbol{s_i}$ to $\boldsymbol{s_j}$ and otherwise $\beta_n^{i,j}=0$. One has

$$\mathcal{C}5:\beta_n^{i,j}\in\{0,1\},n\in\mathcal{N},i,j\in\mathcal{M},i\neq j. \tag{17}$$

In the same manner with [25], the coordinate of UAV standby point is known a priori, denoted by $\boldsymbol{s_0}$ and can be regarded as a special USV without generating task. Since each UAV needs to take off from UAV standby point, one has

$$\mathcal{C}6:\sum_{i\in\mathcal{M}}\beta_n^{0,i}=1,n\in\mathcal{N},i\in\mathcal{M}. \tag{18}$$

Since each UAV needs to fly back to UAV standby point, one has

$$\mathcal{C}7:\sum_{i\in\mathcal{M}}\beta_n^{i,0}=1,n\in\mathcal{N}. \tag{19}$$

Since each USV i can only be served by up to one UAV, one has

$$\mathcal{C}8:\sum_{n\in\mathcal{N}}\sum_{g\in\mathcal{M}}\beta_n^{g,i}=1,i,g\in\mathcal{M},i\neq g. \tag{20}$$

Assume that each UAV n flies at a constant flight speed v_n. The corresponding flying time between any two successive hovering coordinates $\boldsymbol{s_i}$ and $\boldsymbol{s_j}$, can be given as

$$t_n^{i,j}=\frac{\beta_n^{i,j}||\boldsymbol{s_i}-\boldsymbol{s_j}||}{v_n},n\in\mathcal{N},i,j\in\mathcal{M},i\neq j, \tag{21}$$

Let $\tau_{n,i}$ be the arriving time of UAV n to arrive $\boldsymbol{s_i}$, which can be defined as $\tau_{n,i}\triangleq\sum_{g\in\mathcal{M}}\beta_n^{g,i}(\tau_{n,g}+t_{n,g}^w+\alpha_g(t_{b,n,g}+t_g^l)+(1-\alpha_g)(t_{n,g}+t_n^g)+t_n^{g,i}),n\in\mathcal{N},i,g\in\mathcal{M},i\neq g$. The waiting time of UAV n to serve USV i can be mathematically expressed as

$$t_{n,i}^w=\max\{X_i-\tau_{n,i},0\},i\in\mathcal{M}. \tag{22}$$

Since each U_i should be completed not late than Y_i, one has

$$\begin{aligned}\mathcal{C}9:&\tau_{n,i}+t_{n,i}^w+\alpha_i(t_{b,n,i}+t_i^l)+(1-\alpha_i)(t_{n,i}+t_i^n)\leq Y_i,\\&i\in\mathcal{M},n\in\mathcal{N}.\end{aligned} \tag{23}$$

Since cached data size of each UAV n cannot exceed the maximum caching capability D_n^{max}, one has

$$\mathcal{C}10:\sum_{i,j\in\mathcal{M},i\neq j}(1-\alpha_i)\beta_n^{i,j}D_i^o\leq D_n^{max},n\in\mathcal{N}. \tag{24}$$

C. Problem Formulation

In this paper, we aim to minimize USVs bidirectional computation task processing time cost by jointly considering USVs execution selection indicator $\boldsymbol{\alpha}=\{\alpha_i,i\in\mathcal{M}\}$, RIS phase shift vector $\boldsymbol{\theta}=\{\boldsymbol{\theta_{n,i}},i\in\mathcal{M},n\in\mathcal{N}\}$, UAVs flight route selection indicator $\boldsymbol{\beta}=\{\beta_n^{i,j},n\in\mathcal{N},i,j\in\mathcal{M},i\neq j\}$ and UAVs hovering coordinates $\boldsymbol{s}=\{\boldsymbol{s_i},i\in\mathcal{M}\}$, which can be mathematically formulated as

$$\begin{aligned}\mathcal{P}1:&\min_{\boldsymbol{\alpha},\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{s}}\sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{M}}\alpha_i(t_{b,n,i}+t_i^l)+(1-\alpha_i)(t_i^n+t_{i,n})\\s.t.\quad&\mathcal{C}1-\mathcal{C}10.\end{aligned} \tag{25}$$

Note: One can observe that $\mathcal{P}1$ is a mixed-integer non-linear optimization problem and is challenging to solve due to the following reasons. First, due to the existence of 0-1 variables, e.g., $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, the traditional highly efficient nature-inspired algorithms, such as genetic algorithm, differential evolution, particle swarm optimization and so forth, cannot efficiently solve $\mathcal{P}1$ even suffering significantly time cost [3]. Although numerous learning-based optimization approaches, such as deep reinforcement learning-based algorithm [26] and multi-agent reinforcement learning-inspired algorithm [27], have been investigated to enhance RIS-assisted data transmission performance. However, these solutions may cannot realize convergence even suffering huge time cost and computing resources since $\boldsymbol{s}$ and $\boldsymbol{\theta}$ are tightly coupled.

III. The Proposed Solution

To tackle $\mathcal{P}1$, a heuristic solution is proposed, which is illustrated in Fig. 2. $\mathcal{P}1$ can be divided into three subproblems, e.g., UAVs flight route selection indicator subproblem $\mathcal{P}1.1$, USVs execution selection subproblem $\mathcal{P}1.2$ and the joint UAVs hovering coordinates and RIS phase shift vector sub-problem $\mathcal{P}1.3$. After initializing $\boldsymbol{\alpha_0}$, $\boldsymbol{\beta_0}$ and $\boldsymbol{s_0}$, the proposed EDA algorithm performs the matching operation between each UAV n and USV i according to the service waiting time and hovering coordinates of each UAV n; when EDA algorithm reaches convergence, the optimized service set $\mathcal{M}_n$ of each UAV n can be obtained and the order of each element in $\mathcal{M}_n$ can be utilized to indicate the service sequence. As such, the updated $\boldsymbol{\beta}$ can be obtained. Then, the proposed ELRM algorithm divides $\mathcal{P}1.2$ into N parallel sub-problem according to the obtained N service sets of N UAVs. Consider each sub-problem, ELRM algorithm dynamically select execution mode for each USV i in $\mathcal{M}_n$ according to caching capability of each UAV n and time window constraint of each USV i; when the proposed ELRM algorithm reaches convergence, the updated $\boldsymbol{\alpha}$ can be obtained according to a list of parallel execution mode selection subproblems. Subsequently, the proposed ADMM-SCA-based algorithm divides $\mathcal{P}1.3$ into N parallel sub-problems according to N service sets of each UAV n. Consider each sub-problem, ADMM-SCA-based algorithm optimizes the hovering coordinate of each USV i in $\mathcal{M}_n$ in an iterative manner. In particular, there exists two conditions, where $\alpha_i=1$ and $\alpha_i=0$ represent USV i selects local execution mode and UAV execution mode, respectively. Consider **Case 1**, USV i selects local execution mode, ADMM is utilized to optimize hovering coordinate of USV i and the corresponding RIS phase shift vector $\boldsymbol{\theta_{n,i}}$ can be obtained according to **Proposition 1**. Consider **Case 2**, USV i selects UAV execution mode, SCA is utilized to optimize the hovering coordinate of USV i. When the ADMM-SCA-based algorithm reaches convergence, the updated $\boldsymbol{s}$ can be obtained. In this way, one can obtain the feasible solution to the challenging problem $\mathcal{P}1$. The detailed information regarding the proposed solution is listed as follows.

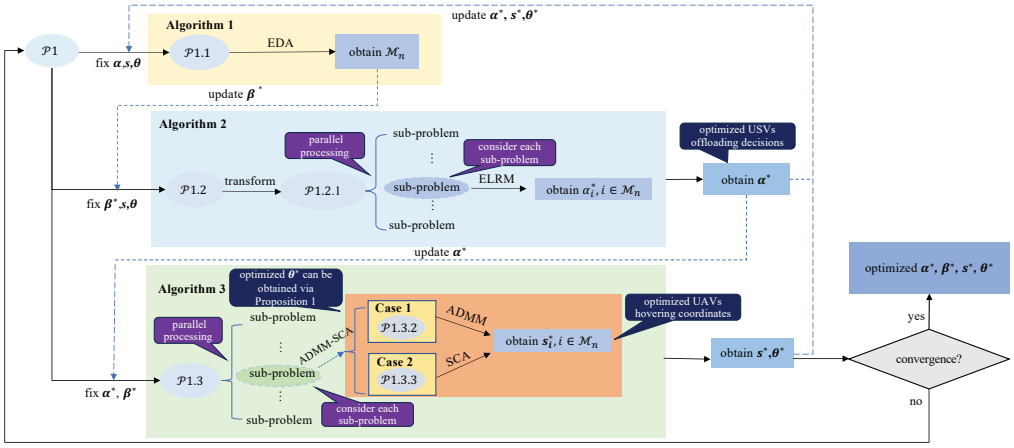

Fig. 2: The overall process to solve $\mathcal{P}1$.

A. The optimization problem of UAVs flight route selection indicator $\boldsymbol{\beta}$

Given any feasible $\boldsymbol{\alpha}$, $\boldsymbol{s}$ and $\boldsymbol{\theta}$, $\mathcal{P}1$ can be reduced as

$$\begin{aligned} &\mathcal{P}1.1: \min_{\boldsymbol{\beta}} \sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{M}} \alpha_i(t_{b,n,i}+t_i^l)+(1-\alpha_i)(t_i^n+t_{i,n}) \\ &s.t. \qquad\qquad \mathcal{C}5-\mathcal{C}10. \end{aligned} \tag{26}$$

Note that the dimensions of UAVs flight route selection indicator $\boldsymbol{\beta}$ can reach up to $M!$, which makes $\mathcal{P}1.1$ extremely difficult to tackle. Moreover, the existance of $\mathcal{C}10$ makes it difficult to apply efficient Ant Colony Algorithms (ACO) to solve $\mathcal{P}1.1$. Fortunately, inspired by [28], we notice that the feasible region of $\mathcal{P}1.1$ can be significantly decreased by transforming $\mathcal{P}1.1$ into a stable marriage problem (SMP), which is proved that can be efficiently solved by utilizing Deferred Acceptance (DA) algorithm. However, the traditional DA algorithm can only apply to one-to-one matching scenarios, which cannot be directly utilized to solve $\mathcal{P}1.1$. In this paper, an EDA algorithm is proposed to tackle $\mathcal{P}1.1$, where the detailed information can be found in **Algorithm 1**. The significant steps of the proposed EDA algorithm are summarized as follows.

Algorithm 1: The Proposed EDA Algorithm

```
initialize M_n = ∅, U_i and S;
while r_EDA ≤ r_EDA^max or S ≠ ∅ do
    UAVs service preference list update
    for i ∈ S do
        for n ∈ N do
            if UAV n satisfies C9 and C10 then
                move this UAV to U_i
            end
        end
    end
    perform matching operation and update M_n
    unserved USV set update
    for i ∈ S do
        if USV i matched one UAV then
            remove USV i from S
        end
    end
    r = r + 1
    until convergence
end
```

USVs service preference list update: Let $\mathcal{N}_i$ and $\mathcal{S}$ be the preference list of each USV i and the set of unserved USVs, respectively. According to Steps 4-10, each USV $i \in \mathcal{S}$ can update its preference list $\mathcal{U}_i$. In particular, each UAV n that can satisfy $\mathcal{C}9$ and $\mathcal{C}10$ can be regarded as a potential service provider for USV $i \in \mathcal{S}$. Then, one can move UAV n into $\mathcal{N}_i$ and sort elements in $\mathcal{N}_i$ in ascending order of $t_{n,i}^w$.

Matching operation Let $\mathcal{M}_n$ be the service set of each UAV n, where the order of each element can be utilized to indicate the service sequence. USV $i \in \mathcal{S}$ that can satisfy $\mathcal{C}9$ and $\mathcal{C}10$ can be regarded as a candidate that can be served by UAV n. Next, the matching operation is utilized. In particular, if UAV n cannot match any USV $i \in \mathcal{S}$, this UAV accepts USV i. When UAV n matches one USV, e.g., USV j, if $d_{n,i} \geq d_{n,j}$, UAV n will reject USV i and then this USV i require matching the next element of $\mathcal{N}_i$. Otherwise, UAV n accepts USV i and then replace USV j by USV i in $\mathcal{M}_n$. The match operation will continue until all USVs have matched one UAV and then move the matched USV i to $\mathcal{M}_n$. The proposed EDA algorithm can be regarded as convergence when $\mathcal{S} = \emptyset$ or $r_{EDA} = r_{EDA}^{max}$, where r_{EDA}^{max} represents the maximum number of iterations of the proposed EDA algorithm. As such, one can obtain the optimal USVs execution selection indicator $\boldsymbol{\beta}^*$.

B. The optimization problem of USVs execution mode selection indicator $\boldsymbol{\alpha}$

After obtain the feasible $\boldsymbol{\beta}$, given any feasible $\boldsymbol{\alpha}$ and $\boldsymbol{s}$, $\mathcal{P}1$ can be reduced as

$$\begin{aligned} &\mathcal{P}1.2: \min_{\boldsymbol{\alpha}} \sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{M}_n} \alpha_i(t_{b,n,i}+t_i^l)+(1-\alpha_i)(t_{i,n}+t_i^n) \\ &s.t. \qquad\qquad \mathcal{C}1, \mathcal{C}9, \mathcal{C}10. \end{aligned} \tag{27}$$

$\mathcal{P}1.2$ is still challenging to tackle due to the following reasons. First, the arrival time of each assigned UAV cannot be determined. Second, the feasible dimensions of USV execution selection indicators can be up to 2^M. As such, the computation time of traditional searching algorithms such as branch and bound becomes inefficient to solve $\mathcal{P}1.2$. Note that $\mathcal{P}1.2$ can be divided into a list of parallel sub-problems and efficiently solve each subproblem via Lagrangian relaxation method. In this way, one can obtain the feasible solution to the challenging optimization problem $\mathcal{P}1.2$ efficiently.

Let $t_i^s = t_{b,n,i} + t_i^l - t_{i,n} - t_i^n$ and $\hat{t}_i^s = Y_i - \tau_{n,i} - t_{n,i}^w - t_{n,i} - t_i^n$. Consider the n-th sub-problem of $\mathcal{P}1.2$, one has

$$\begin{aligned} &\mathcal{P}1.2.1 : \min_{\boldsymbol{\alpha}} \sum_{i\in\mathcal{M}_n} (\alpha_i t_i^s + t_{i,n} + t_i^n) \\ &s.t. \qquad\qquad \mathcal{C}1, \\ &\quad \dot{\mathcal{C}}9 : \alpha_i t_i^s \le \hat{t}_i^s, i \in \mathcal{M}_n, \\ &\quad \dot{\mathcal{C}}10 : \sum_{i\in\mathcal{M}_n} (1-\alpha_i) D_i^o \le D_n^{max}. \end{aligned} \tag{28}$$

In this paper, an ELRM algorithm is proposed to solve $\mathcal{P}1.2.1$. Let $\boldsymbol{\lambda}$ and μ be the introduced non-negative Lagrange multipliers, where $\boldsymbol{\lambda} \triangleq \{\lambda_1, \lambda_2, \cdots, \lambda_M\}$. The Lagrange function $LR(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \mu)$ can be formulated as

$$\begin{aligned} LR(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \mu) &= \sum_{i\in\mathcal{M}_n} \alpha_i t_i^s + t_{i,n} + t_i^n + \sum_{i\in\mathcal{M}_n} \lambda_i(\alpha_i t_i^s - \hat{t}_i^s) \\ &+ \mu(\sum_{i\in\mathcal{M}_n} (1-\alpha_i) D_i^o - D_n^{max}), \end{aligned} \tag{29}$$

In this way, $\mathcal{P}1.2.1$ can be rewritten as

$$\begin{aligned} \hat{\mathcal{P}}1.2.1 : &\max_{\boldsymbol{\lambda},\mu} \min_{\boldsymbol{\alpha}} \sum_{i\in\mathcal{M}_n} (t_i^s + \lambda_i t_i^s - \mu D_i^o)\alpha_i \\ &+ \sum_{i\in\mathcal{M}_n} (t_{i,n} + t_i^n - \lambda_i \hat{t}_i^s + \mu D_i^o) - \mu D_n^{max} \\ s.t. \qquad &\lambda_i \ge 0, \mu \ge 0. \end{aligned} \tag{30}$$

Define $\boldsymbol{\alpha}^{r*}(\boldsymbol{\lambda}^r, \mu^r)$ as the obtained optimal solution in r-th iteration. One has

$$\alpha_i^{r*}(\boldsymbol{\lambda}^r, \mu^r) = \begin{cases} 1, & t_i^s + \lambda_i t_i^s - \mu D_i^o \le 0, \\ 0, & \text{otherwise.} \end{cases} \tag{31}$$

Substitute $\boldsymbol{\alpha}^{(r-1)*}(\boldsymbol{\lambda}^{r-1}, \mu^{r-1})$ into $\hat{t}_i^s$, one can obtain $\bar{t}_i^s$, which is a constant. Given any feasible $\boldsymbol{\alpha}^{r*}(\boldsymbol{\lambda}^r, \mu^r)$, $\hat{\mathcal{P}}1.2.1$ can be reduced as

$$\begin{aligned} \dot{\mathcal{P}}1.2.1 : &\max_{\boldsymbol{\lambda},\mu} \sum_{i\in\mathcal{M}_n} (t_i^s + \lambda_i t_i^s - \mu D_i^o)\alpha_i^{r*} \\ &+ \sum_{i\in\mathcal{M}_n} (t_{i,n} + t_i^n - \lambda_i \bar{t}_i^s + \mu D_i^o) - \mu D_n^{max} \\ s.t. \qquad &\lambda_i \ge 0, \mu \ge 0. \end{aligned} \tag{32}$$

The subgradient method is utilized to update Lagrange multipliers in each iteration, where λ_i^{r+1} can be given as

$$\lambda_i^{r+1} = \max\{\lambda_i^r + \omega_i^r(\alpha_i^{r*} t_i^s - \bar{t}_i^s), 0\}, \tag{33}$$

where $\omega_i^r \in (0,1)$ represents the proportionality coefficient of λ_i^{r+1}. μ^{r+1} can be expressed as

$$\mu^{r+1} = \max\{\mu + \phi_i^r(\sum_{i\in\mathcal{M}_n} (1-\alpha_i^{r*}) D_i^o - D_n^{max}), 0\}, \tag{34}$$

where $\phi_i^r \in (0,1)$ is the proportionality coefficient of μ^{r+1}. Let ϵ and r_{ELRM}^{max} be the error control parameter and the maximum number of iterations of the proposed ELRM algorithm, respectively. The proposed ELRM algorithm reaches convergence when $||\boldsymbol{\lambda}^{r+1} - \boldsymbol{\lambda}^r|| \le \epsilon$ or $r_{ELRM} = r_{ELRM}^{max}$. As such, one can obtain the optimal USVs execution mode selection indicator $\boldsymbol{\alpha}^*$. The detailed information regarding the proposed ELRM algorithm is summarized in **Algorithm 2**.

Algorithm 2: The proposed ELRM algorithm

```
divide P1.2 into N parallel sub-problems
consider each sub-problem
for n ∈ 𝒩 do
    initialize λ , μ and ϵ
    construct Lagrange function LR(α, λ, μ)
     according to Eq. (29)
    while r_ELRM ≤ r_ELRM^max or ||λ_{r+1} − λ_r|| ≥ ϵ do
        update t̂_i^s
        update λ and μ according to Eq. (33)-(34)
        obtain α^{r*}(λ^r, μ^r)
    end
    until convergence
end
```

C. The joint optimization subproblem of UAVs hovering coordinates $\boldsymbol{s}$ and RIS phase shift vector $\boldsymbol{\theta}$

After obtain the feasible $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, $\mathcal{P}1$ can be reduced as

$$\begin{aligned} &\mathcal{P}1.3 : \min_{\boldsymbol{\theta},\boldsymbol{s}} \sum_{n\in\mathcal{N}} \sum_{i\in\mathcal{M}_n} \alpha_i^*(t_{b,n,i} + t_i^l) + (1-\alpha_i^*)(t_{i,n} + t_i^n) \\ &s.t. \qquad\qquad \mathcal{C}2 - \mathcal{C}4, \mathcal{C}8. \end{aligned} \tag{35}$$

One can observe that $\boldsymbol{s}$ and $\boldsymbol{\theta}$ are closely coupled in $t_{b,n,i}$ and the denominator of $t_{b,n,i}$ cannot be simplified due to the existence of $\boldsymbol{\theta_{n,i}}$. Moreover, the objective function of $\mathcal{P}1.3$ and $\mathcal{C}9$ are both non-convex with respect to $\boldsymbol{s_i}$. As such, $\mathcal{P}1.3$ cannot be directly solved by using the traditional convex optimization algorithm and learning-based solutions.

Note that $\mathcal{P}1.3$ can be divided into a list of parallel sub-problems. Consider each sub-problem, given any feasible $\boldsymbol{s}$, after remove irrelevant parameters, $\mathcal{P}1.3$ can be reduced as

$$\begin{aligned} &\hat{\mathcal{P}}1.3.1 : \min_{\boldsymbol{\theta}} \sum_{i\in\mathcal{M}_n} \alpha_i^*(t_{b,n,i} + t_i^l) + (1-\alpha_i^*)(t_{b,i} + t_i^n) \\ &s.t. \qquad\qquad \mathcal{C}2. \end{aligned} \tag{36}$$

According to Eqs. (8)-(9), after remove irrelevant terms, $\hat{\mathcal{P}}1.3.1$ can be rewritten as a maximization problem as follows

$$\begin{aligned} &\widetilde{\mathcal{P}}1.3.1 : \max_{\boldsymbol{\theta}} \sum_{i\in\mathcal{M}_n} ||\boldsymbol{w_i^H}(\boldsymbol{h_{b,i}} + \boldsymbol{h_{b,n}\Theta_{n,i}h_{n,i}})||^2 \\ &s.t. \qquad\qquad \mathcal{C}2. \end{aligned} \tag{37}$$

According to **Proposition 1**, one can obtain the the optimal solution to $\widetilde{\mathcal{P}}1.3.1$.

Proposition 1: The optimal solution to $\widetilde{\mathcal{P}}1.3.1$ can be given as $\boldsymbol{\theta^*_{n,i}} = \{\theta^{1*}_{n,i}, \theta^{2*}_{n,i}, \cdots, \theta^{k*}_{n,i}, \cdots, \theta^{K*}_{n,i}\}$, where each $\theta^{k*}_{n,i} = \arg(\Psi_i) - \arg(\boldsymbol{\Omega_i}(k))$.

Proof. The proof can be found in [29]. □

After successfully solving $\widetilde{\mathcal{P}}1.3.1$, one can optimize UAVs hovering coordinates in an iterative manner. Since α_i^* can be obtained using **Algorithm 2**, this section investigates the optimization of UAVs hovering coordinates under two cases, e.g., $\alpha_i^* = 1$ and 0, respectively. The detailed information is given as follows.

Case 1: When $\alpha_i^* = 1$, given any feasible $\boldsymbol{s_g}, i \neq g$ and $\boldsymbol{\theta^*_{n,i}}$, $t_{b,n,i}$ can be rewritten as

$$t_{b,n,i} = \frac{D_i^o}{B_{b,n,i}\log_2(1 + (\frac{\rho_0}{||\boldsymbol{s_i} - \boldsymbol{q_i}||||\boldsymbol{s_i} - \boldsymbol{q_b}||} + \sqrt{\rho_0 d_{b,i}^{-3.5}})^2)}. \tag{38}$$

Consequently, $\mathcal{P}1.3$ can be transformed into

$$\begin{aligned} &\mathcal{P}1.3.2 : \min_{\boldsymbol{s_i}} \ t_{b,n,i} + t_i^l \\ &s.t. \qquad \mathcal{C}3, \mathcal{C}4, \mathcal{C}9. \end{aligned} \tag{39}$$

One can observe that the objective function and constraint $\mathcal{C}9$ of $\mathcal{P}1.3.2$ are both non-convex, which makes $\mathcal{P}1.3.2$ still challenging to solve.

Define the auxiliary variables $r_{b,n,i}, n \in \mathcal{N}, i \in \mathcal{M}_n$ and $\Gamma_i = Y_i - \frac{D_i^o}{B_{b,n,i} r_{b,n,i}} - t_i^l - X_i - \sum_{g \in \mathcal{M}} \beta_n^{g,i*}(\tau_{n,g} - \alpha_g^*(t_{b,n,g} + t_g^l) - (1 - \alpha_g^*)(t_{n,g} + t_g^n))$, where α_g^* is the optimal execution selection indicator of USV $g, g \in \mathcal{M}_n$. In this way, $\mathcal{P}1.3.2$ can be transformed into

$$\begin{aligned} &\hat{\mathcal{P}}1.3.2 : \min_{\boldsymbol{s_i}, r_{b,n,i}} \ t_{b,n,i} + t_i^l \\ &s.t. \qquad \mathcal{C}3, C4 \\ &\quad \hat{\mathcal{C}}9 : \ ||\boldsymbol{s_i} - \boldsymbol{s_g}|| \leq v_n \Gamma_i, i, j \in \mathcal{M} \\ &\quad \mathcal{C}11 : \ r_{b,n,i} \leq C_{b,n,i}, i \in \mathcal{M}, n \in \mathcal{N}. \end{aligned} \tag{40}$$

Note that the traditional high efficient solutions such as SCA cannot be directly utilized to transform $\mathcal{C}11$ into a convex constraint since $||\boldsymbol{s_i} - \boldsymbol{q_i}||$ and $||\boldsymbol{s_i} - \boldsymbol{q_b}||$ are closely coupled.

To tackle the challenging problem $\hat{\mathcal{P}}1.3.2$, ADMM is utilized. After introduce the duplicated variable $\hat{\boldsymbol{s_i}}, i \in \mathcal{M}_n$, one has

$$\mathcal{C}12 : \boldsymbol{s_i} = \hat{\boldsymbol{s_i}}, i \in \mathcal{N}_i. \tag{41}$$

Consequently, $\mathcal{C}4$ can be transformed into

$$\hat{\mathcal{C}}4 : ||\hat{\boldsymbol{s_i}} - \boldsymbol{q_b}|| \leq d_{b,n}^{max}, i \in \mathcal{M}_n, \tag{42}$$

$\mathcal{C}11$ can be transformed into

$$\hat{\mathcal{C}}11 : r_{b,n,i} \leq \log_2(1 + (\frac{\rho_0}{||\boldsymbol{s_i} - \boldsymbol{q_i}||||\hat{\boldsymbol{s_i}} - \boldsymbol{q_b}||} + \sqrt{\rho_0 d_{b,i}^{-3.5}})^2). \tag{43}$$

In this way, $\hat{\mathcal{P}}1.3.2$ can be transformed into

$$\begin{aligned} &\dot{\mathcal{P}}1.3.2 : \min_{\boldsymbol{s_i}, \hat{\boldsymbol{s_i}}, r_{b,n,i}} \ t_{b,n,i} + t_i^l \\ &s.t. \qquad \mathcal{C}3, \hat{\mathcal{C}}4, \hat{\mathcal{C}}9, \hat{\mathcal{C}}11, \mathcal{C}12. \end{aligned} \tag{44}$$

The corresponding Augmented Lagrangian (AL) function of $\hat{\mathcal{P}}1.3.2$ can be formulated as

$$\mathcal{L}(\boldsymbol{s_i}, \hat{\boldsymbol{s_i}}, \boldsymbol{\vartheta_i}) = t_{b,n,i} + \frac{\rho}{2}||\boldsymbol{s_i} - \hat{\boldsymbol{s_i}} + \boldsymbol{\vartheta_i}||_F^2, \tag{45}$$

where ρ is the penalty parameter and $\boldsymbol{\vartheta_i}$ is the dual variable of $\mathcal{C}12$. One can observe that $\mathcal{L}(\boldsymbol{s_i}, \hat{\boldsymbol{s_i}}, \boldsymbol{\vartheta_i})$ is separable along with $\boldsymbol{s_i}$ and $\hat{\boldsymbol{s_i}}$. Therefore, the feasible solution to $\hat{\mathcal{P}}1.3.2$ can be obtained by alternately updating $\boldsymbol{s_i}$ and $\hat{\boldsymbol{s_i}}$. The detailed information is summarized as follows.

The Update of $\hat{\boldsymbol{s_i}}$: The optimization of $\hat{\boldsymbol{s_i}}$ in each r-th iteration can be given as

$$\begin{aligned} &\mathcal{P}1.3.2.1 : \min_{\hat{\boldsymbol{s_i}}} \ \frac{\rho}{2}||\hat{\boldsymbol{s_i}} - \boldsymbol{s_i}^{r-1} + \boldsymbol{\vartheta_i}||_F^2, \\ &s.t. \qquad \hat{\mathcal{C}}4, \hat{\mathcal{C}1}1, \end{aligned} \tag{46}$$

where $\boldsymbol{s_i^{r-1}}$ is the obtained solution of $\boldsymbol{s_i}$ in $(r-1)$-th iteration. $\hat{C}_{b,n,i}$ is convex with respect to $||\hat{\boldsymbol{s_i}} - \boldsymbol{q_b}||$ and thus $\hat{\mathcal{C}1}1$ can be transformed into a convex constraint by utilize SCA method. The lower bound of $\hat{C}_{b,n,i}$ can be determined by employing the first-order Taylor expansion of $\hat{C}_{b,n,i}$. One can obtain the expression of $\hat{C}^{lb}_{b,n,i}$, which is shown in Eq. (47). where $\Delta = \sqrt{\rho_0 d_{b,i}^{-3.5}}$.

In this way, $\hat{\mathcal{C}1}1$ can be transformed into

$$\dot{\mathcal{C}1}1 : r_{b,n,i} \leq \hat{C}^{lb}_{b,n,i}. \tag{48}$$

Consequently, $\mathcal{P}1.3.2.1$ can be transformed into

$$\begin{aligned} &\hat{\mathcal{P}}1.3.2.1 : \min_{\hat{\boldsymbol{s_i}}} \frac{\rho}{2}||\hat{\boldsymbol{s_i}} - \boldsymbol{s_i}^{r-1} + \boldsymbol{\vartheta_i}||_F^2, \\ &s.t. \qquad \hat{\mathcal{C}}4, \dot{\mathcal{C}1}1, \end{aligned} \tag{49}$$

Note that $\hat{\mathcal{P}}1.3.2.1$ is a convex optimization problem, which can be efficiently solved by CVX [30].

The Update of $\boldsymbol{s_i}$: The optimization of $\boldsymbol{s_i}$ in r-th iteration can be given as

$$\begin{aligned} &\mathcal{P}1.3.2.2 : \min_{\boldsymbol{s_i}} \ t_{b,n,i} + \frac{\rho}{2}||\boldsymbol{s_i} - \hat{\boldsymbol{s_i}}^{r-1} + \boldsymbol{\vartheta_i}||_F^2, \\ &s.t. \qquad \mathcal{C}3, \hat{\mathcal{C}}11, \end{aligned} \tag{50}$$

Although $\hat{\mathcal{C}}11$ can be transformed into a convex constraint by utilizing SCA method, $\mathcal{P}1.3.2.2$ is still a non-convex optimization problem and challenging to solve since $t_{b,n,i}$ is non-convex with respect to $\boldsymbol{s_i}$ [31].

After substitute $t_{b,n,i}$ by $\hat{t}_{b,n,i}$, where $\hat{t}_{b,n,i} = \frac{D_i^o}{B_{b,n,i} r_{b,n,i}}$. Consider each iteration, $r_{b,n,i}$ can be updated by substitute $\boldsymbol{s_i}$ by $\hat{\boldsymbol{s_i}}^r$. In this way, $\dot{\mathcal{P}}1.3.2.2$ can be transformed into

$$\begin{aligned} &\dot{\mathcal{P}}1.3.2.2 : \min_{\boldsymbol{s_i}} \ \hat{t}_{b,n,i} + \frac{\rho}{2}||\boldsymbol{s_i} - \hat{\boldsymbol{s_i}}^{r-1} + \boldsymbol{\vartheta_i}||_F^2, \\ &s.t. \qquad \mathcal{C}3, \\ &\qquad \overline{\mathcal{C}}11 : r_{b,n,i} \leq C^{lb}_{b,n,i}. \end{aligned} \tag{51}$$

One can observe that $\dot{\mathcal{P}}1.3.2.2$ is a convex optimization problem, which can be efficiently solved by using CVX. In this way, one can obtain the feasible solution to **Case 1**. In summary, one can utilize ADMM to divide $\hat{\mathcal{P}}1.3.2$ into $\hat{\mathcal{P}}1.3.2.1$ and $\dot{\mathcal{P}}1.3.2.2$, and then solve each problem using

$$\hat{C}^{lb}_{b,n,i} \triangleq \log_2(1+(\frac{\rho_0}{||\boldsymbol{s_i}^{r-1}-\boldsymbol{q_i}||||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||}+\Delta)^2) - \frac{2(\rho_0^2+\Delta\rho_0||\boldsymbol{s_i}^{r-1}-\boldsymbol{q_i}||||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||)}{\ln 2(||\boldsymbol{s_i}^{r-1}-\boldsymbol{q_i}||^2||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||^3+||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||(\rho_0+\Delta||\boldsymbol{s_i}^{r-1}-\boldsymbol{q_i}||||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||)^2)}(||\hat{\boldsymbol{s_i}}-\boldsymbol{q_b}||-||\hat{\boldsymbol{s_i}}^{r-1}-\boldsymbol{q_b}||), \tag{47}$$

CVX. As such, the proposed ADMM algorithm can realize convergence when reaches the maximum iteration number r^{max}_{ADMM}.

Case 2: When $\alpha_i^*=0$, given any feasible $\boldsymbol{s_j}, j\neq i$, $\mathcal{P}1.3$ can be reduced as

$$\begin{aligned} &\mathcal{P}1.3.3: \min_{\boldsymbol{s_i}} t_{n,i}+t_i^n \\ &s.t. \qquad \mathcal{C}3,\mathcal{C}4,\mathcal{C}9. \end{aligned} \tag{52}$$

After introduce auxiliary variable $z_{i,n}$, $\mathcal{P}1.3.3$ can be transformed into

$$\begin{aligned} &\hat{\mathcal{P}}1.3.3: \min_{\boldsymbol{s_i},z_{i,n}} t_{n,i}+t_i^n \\ &s.t. \qquad \mathcal{C}3,\mathcal{C}4, \\ &\overline{\mathcal{C}}9: \tau_{n,i}+t^w_{n,i}+\frac{D_i^l}{z_{n,i}}+t_i^n\leq Y_i, \\ &\mathcal{C}13: z_{n,i}\leq C_{i,n}. \end{aligned} \tag{53}$$

One can apply SCA to $\mathcal{C}13$ and thus the lower bound of $z_{n,i}$ can be given as

$$\begin{aligned} z^{lb}_{n,i} \triangleq B_{n,i}\log_2(1+\frac{p_i^{tr}h_{i,n}}{(||\overline{\boldsymbol{s_i}}-\boldsymbol{q_i}||^2)\sigma^2}) \\ -\frac{B_{n,i}p_i^{tr}h_{i,n}(||\boldsymbol{s_i}-\boldsymbol{q_i}||^2-||\overline{\boldsymbol{s_i}}-\boldsymbol{q_i}||^2)}{\sigma^2\ln 2||\overline{\boldsymbol{s_i}}-\boldsymbol{q_i}||^2(||\overline{\boldsymbol{s_i}}-\boldsymbol{q_i}||^2+\frac{p_i^{tr}h_{i,n}}{\sigma^2})}. \end{aligned} \tag{54}$$

As such, $\hat{\mathcal{P}}1.3.3$ can be transformed into a convex optimization problem, which can be efficiently solved by CVX. In this way, one can obtain the feasible solution to **Case 2**. The detailed information regarding the proposed joint ADMM-SCA-based algorithm can be found in **Algorithm 3**, where $r^{max}_{ADMM-SCA}$ is the maximum iteration number.

The framework regarding the proposed solution is summarized in **Algorithm 4**. Let r^* be the number of iterations for the proposed solution to realize convergence. In particular, the complexity of solving problem $\mathcal{P}1.1$ is $\mathcal{O}(NM^2)$ and complexity of solving problem $\mathcal{P}1.2$ is $\mathcal{O}(M\log_2\frac{1}{\epsilon})$. Note that when all USVs select local execution mode, **Algorithm 3** reaches the worst situation. The complexity of solving problem $\mathcal{P}1.3$ under the worst situation is $\mathcal{O}(I_{ADMM}M^{1.5}N^{3.5})$. Therefore, the complexity of **Algorithm 4** can be roughly given as $\mathcal{O}(r^*(NM^2+M\log_2\frac{1}{\epsilon}+I_{ADMM}M^{1.5}N^{3.5})$. One should note that MEC server is responsible for executing the proposed solution, where MEC server is equipped with sufficient computation resources and the obtained information can be transmitted to UAV standby point and RIS controller. Moreover, the communication overhead is ignored since it is significantly smaller than computation tasks.

Algorithm 3: The proposed ADMM-SCA-based algorithm

```
divide P1.3 into N parallel sub-problems
consider each subproblem
for n ∈ N do
    for r_{ADMM-SCA} ≤ r^{max}_{ADMM-SCA} do
        Case 1:
        introduce r_{b,n,i}, ŝ_i and C12
        transform C4 into Ĉ4
        transform C11 into Ĉ11
        relax P1.3.2 into ADMM form P̂1.3.2
        while r ≤ r^{max}_{ADMM} do
            solve P̂1.3.2.1 and obtain s_i^g
            solve Ṗ1.3.2.2 and obtain ŝ_i^g
            update Lagrange Multipliers ϑ_i^{g+1}
            update r_{b,n,i}
        end
        Case 2:
        introduce z_{i,n}
        while r ≤ r^{max}_{SCA} do
            solve P̂1.3.3 and obtain s_i^g
            update z_{n,i}
        end
    end
end
```

Algorithm 4: The framework of the proposed solution

```
initial α_0, β_0 and s_0
while r ≤ r* do
    given α_{r-1} and s_{r-1}, solve P1.1 via Algorihm 1 and obtain β_r
    given β_r and s_{r-1}, solve P1.2 via Algorihm 2 to update α_r
    given β_r and α_r, solve P1.3 via Algorihm 3 to update s_r
end
until convergence
```

IV. Performance Evaluation

This section evaluates and compares the selected key performance metrics of the proposed solution with numerous benchmark algorithms. All experiments are conducted in MATLAB with Intel Core i5-12600K CPU @3.70GHz with 16GB RAM. Also, MATLAB with CVX toolbox is also utilized. The simulation setup of the proposed network is given as follows. USVs are randomly distributed in a square area of

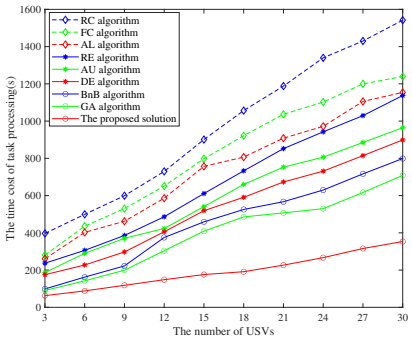

Fig. 3: The task processing time cost versus the different number of USVs when $K = 10$.

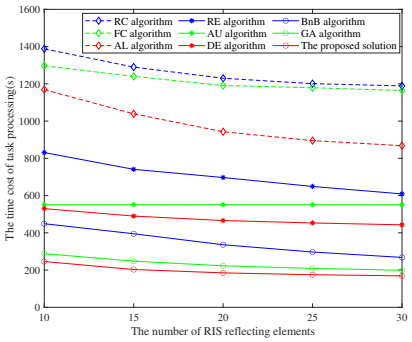

Fig. 4: The task processing time cost versus the different number of RIS reflecting elements when $M = 15$.

300 m $\times$ 300 m and UAVs fly at a fixed altitude of $H = 50$ m. The data size generated by each USV and originated from the Internet are set as [1, 20] $\times 10^7$ bits and [1, 20] $\times 10^6$ bits, respectively. The required CPU cycles to execute each task is set as [1, 20] $\times 10^9$. The earliest starting time X_i and latest service time Y_i range from [1, 500] s and (500, 1000] s. To highlight the efficiency of the proposed solution regarding USVs execution selection, the computation capability of each USV and UAV are set as $f_i = 5 \times 10^8$ CPU cycles/s and $f_n = 2 \times 10^9$ CPU cycles/s, respectively. Wireless communication channels are assumed to be perfectly estimated when utilizing RIS technique, TBS-UAV link is assumed to be a controllable NLOS channel. TBS-USV link and UAV-USV link are assumed as non-LOS and LOS, respectively [32]. The PL coefficients, where TBS $\rightarrow$ USV i, UAV-mounted RIS $\rightarrow$ USV i and TBS $\rightarrow$ UAV-mounted RIS are set to 3.5, 2 and 2, respectively, according to the real-world measurements [33]. A list of selected widely used algorithms, e.g., All Local Execution algorithm (AL), All UAVs Execution algorithm (AU), Random Execution algorithm (RE), Genetic algorithm (GA), Branch and Bound algorithm (BnB), UAVs Fixed Hovering Coordinate algorithm (FC), UAVs Random Hovering Coordinates algorithm (RC), Differential Evolution optimization algorithm (DE) are selected as benchmarks. The performance evaluation is investigated via two key performance metrics, e.g., the time cost of task processing and task success execution ratio. Moreover, the performance of the proposed solution under different typical number of deployed UAVs and caching capabilities is explored. The detailed information regarding the selected algorithms is introduced as follows.

All USVs Local Execution algorithm (AL): All USVs execute tasks by local execution mode [3]. The optimization of UAVs flight route selection, RIS phase shift vector and UAVs hovering coordinates is identical to the proposed solution. In this case, even though USVs task execution energy consumption is high, this algorithm can realize considerable information security.

All UAVs Execution algorithm (AU): All USVs decide to execute tasks via UAV execution mode [34]. The optimization of UAVs flight route selection, RIS phase shift vector and UAVs hovering coordinates is identical to the proposed solution.

Random Execution algorithm (RE): All USVs randomly select execution mode [33]. The optimization process of UAVs flight route selection, RIS phase shift vector and UAVs hovering coordinates is identical to the proposed solution.

Genetic algorithm (GA): The optimization of USVs execution mode selection is in the same manner with [35]. The optimization process of RIS phase shift vector and UAVs hovering coordinates is identical to the proposed solution.

Branch and Bound algorithm (BnB): The branch and bound algorithm is utilized to optimize USVs execution mode selection [36]. The optimization process of RIS phase shift vector and UAVs hovering coordinates is identical to the proposed solution.

UAVs Fixed Hovering Coordinate algorithm (FC): Each UAV hovering coordinate is predetermined as the midpoint between TBS and each USV [29]. The optimization of UAVs flight route selection, USVs task execution mode selection and RIS phase shift vector is identical to the proposed solution.

UAVs Random Hovering Coordinates algorithm (RC): UAVs hovering coordinates are randomly determined [37]. The optimization of UAVs flight route selection, USVs task execution mode selection and RIS phase shift vector is identical to the proposed solution.

Differential Evolution optimization algorithm (DE): The differential evolution optimization algorithm is utilized to optimize UAVs hovering coordinates as proposed in [7]. The optimization of UAVs flight route selection, USVs task execution mode selection and RIS phase shift vector is identical to the proposed solution.

The time cost of task processing versus the number of USVs when RIS reflecting elements $K = 10$ is demonstrated in Fig. 3. One can observe that the task processing time cost keeps increasing with the number of USVs rising. In particular, the proposed solution realizes the least processing time at nearly 210 s and 360 s when $M = 15$ and 30, respectively, followed by GA and BnB with the corresponding value of 420 s, 700 s and 490 s, 800 s, respectively. RC realizes the worst performance, where the task processing time cost is approximately 870 s and 1540 s when $M = 15$ and 30, respectively. The reason may involve the fact that the proposed solution can obtain the optimized UAVs hovering coordinates

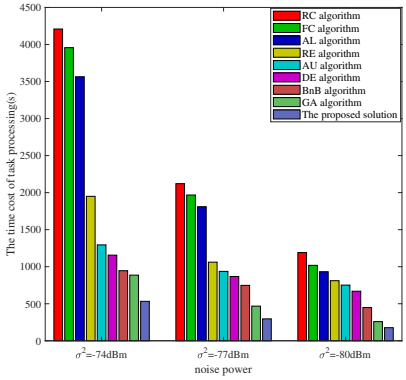

Fig. 5: The task processing time cost versus noise power when $M = 15$.

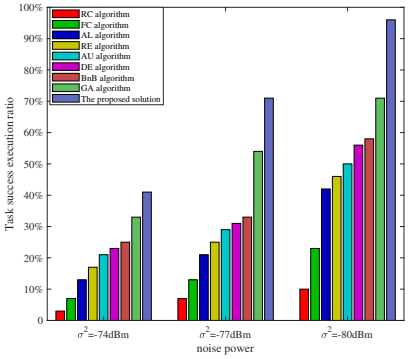

Fig. 6: Task success execution ratio versus noise power.

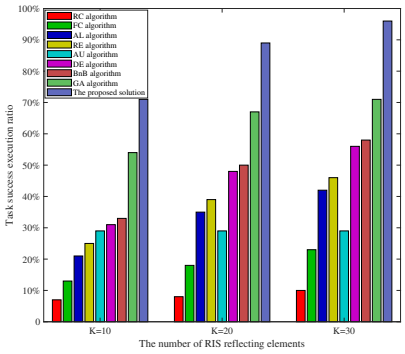

Fig. 7: Task success execution ratio versus the different typical number of RIS reflecting elements.

and consequently reduce transmission distance between USVs and UAVs with enhanced data link quality. Moreover, one should note that appropriate USVs execution selection can remarkably decrease task processing time cost, where the proposed solution can dynamically adjust task execution mode according to the processing time of each USV in comparison with the selected benchmark algorithms. Although GA and BnB are capable of optimizing USVs execution selection, these algorithms may become inefficient in obtaining appropriate USVs execution mode selection without optimizing RIS phase shift vector, UAVs flight route selection indicator and UAVs hovering coordinates.

The typical value of $M = 15$ is selected for further investigation. The task processing time cost versus the different number of RIS reflecting elements when $M = 15$ is demonstrated in Fig. 4. One can observe that the task processing time keeps decreasing with the number of RIS reflecting elements increasing. The proposed solution realizes the best performance, where the processing time is approximately 220 s and 180 s when $K = 10$ and 30, respectively, followed by GA and BnB, where the corresponding value is around 280 s, 200 s and 450 s, 290 s, respectively. RC realizes the worst performance with the task processing time at approximately 1390 s and 1270 s when $K = 10$ and 30, respectively. This is because a larger number of RIS reflecting elements can enhance communication link quality between TBS and each USV and thus decrease transmission latency. Moreover, one should be aware that deploying more RIS reflecting elements cannot considerably improve the link quality between each UAV and USV since a large proportion of task processing time cost is caused by USVs that select UAV execution mode.

The relationship between task processing time cost and the different noise power is shown in Fig. 5. One can observe that the task processing time cost of all algorithms keeps increasing as noise power rises. In particular, the proposed solution realizes the least processing time cost at nearly 200 s, 250 s and 500 s when $\sigma^2 = -80$ dBm, -77 dBm and -74 dBm, respectively, followed by GA and BnB at approximately 250 s, 490 s, 920 s and 490 s, 750 s, 980 s, respectively. RC realizes the worst performance with the corresponding task processing time cost at around 1200 s, 2100 s and 4200 s when $\sigma^2 = -80$ dBm, -77 dBm and -74 dBm, respectively. This may involve the following reasons. First, the larger predetermined σ^2 leads to a lower SNR requirement and thus the transmission time cost between each UAV and USV, and TBS and each USV can be significantly prolonged. Moreover, one should note that inappropriate USVs execution selection may degrade the transmission data rate; the proposed solution can dynamically select an execution mode under different noise power conditions compared with the selected algorithms.

The typical value of $K = 10$ is selected for further investigation. Note that each task that cannot be successfully executed within its time window can be regarded as failed. Following [3], task execution success ratio, which can be defined as the percentage of successfully executed tasks, is taken into consideration. Fig. 6 demonstrates task execution success ratio versus the different noise power. One can observe that task execution success ratio of all algorithms keeps increasing correspondingly as noise power decreases. In particular, the proposed solution realizes the highest task execution success ratio at about 59%, 89% and 94% when $\sigma^2 = -74$dBm, -77dBm and -80dBm, respectively, followed by DE and FC with the corresponding values at about 23%, 32%, 56% and 7%, 15%, 23%, respectively. RC reaches

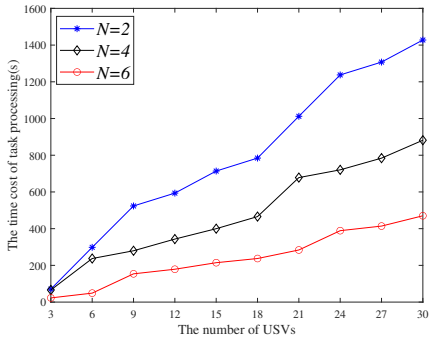

Fig. 8: The task processing time versus the different typical number of UAVs when $D_n^{max} = 10^9$ bits.

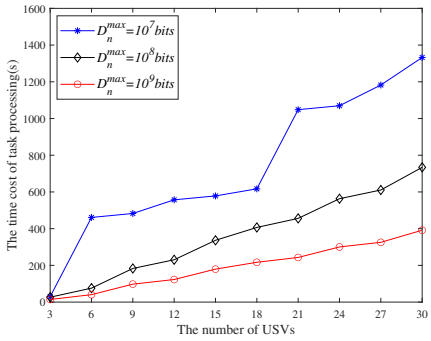

Fig. 9: The task processing time versus the different typical UAVs caching capabilities.

the worst performance with the task execution success ratio of 3%, 9% and 13% when $\sigma^2 = -74$dBm, -77dBm and -80dBm, respectively. This may involve the following facts. First, one should note that lower σ^2 may lead to higher SNR and thus the transmission time cost can decrease significantly. Moreover, the proposed solution can dynamically optimize UAVs hovering coordinates according to task time window constraints. In this way, the waiting time cost of each UAV and task execution time cost can be significantly reduced. Besides, the proposed solution outperforms GA and BnB since the proposed ELRM algorithm can dynamically update Lagrange multipliers when considering $\mathcal{C}9$ while GA and BnB can only utilize the fixed step as analyzed in [35-36].

Fig. 7 demonstrates task execution success ratio versus the different typical number of RIS reflecting elements when $\sigma^2 = -77$ dBm. One can see that the proposed solution outperforms other algorithms when under the same number of K. In particular, task success execution ratio of the proposed solution realizes nearly 70%, 85% and 95% when $K = 10$, 20, and 30, respectively, followed by GA and BnB with the corresponding values at about 55%, 65%, 72% and 33%, 48%, 58%, respectively. RC reaches the worst performance with task execution success ratio of 6%, 8% and 10% when $K = 10$, 20, and 30, respectively. Moreover, one should note that the performance of AU cannot always keep increasing with the number of RIS reflecting elements rising. This is because deploying a larger number of RIS reflecting elements cannot enhance task execution success ratio of AU since this algorithm promises all USVs select UAV execution mode and do not suffer downlink data transmission time cost.

Fig. 8 demonstrates the time cost of task processing versus the number of USVs under different typical number of UAVs. One can observe that the higher number of deployed UAVs promises less task processing time cost. In particular, when $N = 6$, the task processing time cost realizes around 210 s and 440 s when $M = 15$ and 30, respectively, while the corresponding value is nearly 410 s and 890 s when $N = 4$. The reason may involve the fact that when the number of deployed UAVs is small, on the one hand, each UAV needs to serve more USVs during each service duration; on the other hand, to satisfy time window constraint, the distance between two successive UAVs hovering coordinates is prolonged, as analyzed in the proposed ADMM-SCA-based algorithm. Also, one should note that a smaller number of deployed UAVs may bring inadequate caching capability and thus deteriorate task processing time cost.

Fig. 9 demonstrates the time cost of task processing versus the number of USVs under different UAVs caching capabilities of the proposed solution. One can observe that the higher UAVs caching capability promises less task processing time cost. In particular, when $D_n^{max} = 10^9$ bits, the task processing time cost is about 200 s and 400 s when $M = 15$ and 30, respectively, while this value is approximately 380 s and 750 s when $D_n^{max} = 10^8$ bits, respectively. This may involve the reason that USVs compete UAVs caching resources as analyzed in ELRM algorithm and the situation may become even worse when UAVs are equipped with insufficient caching capability, which makes USVs unable to select appropriate execution modes and thus prolong the task processing time cost. For instance, when UAVs suffer limited insufficient caching capabilities, e.g., when $D_n^{max} = 10^7$ bits, the task processing time may result in a sharp increase since some USVs can only be able to select local execution mode and experience high data transmission latency, especially when $M = 6$ and 21.

Note that some existing highly efficient solutions, such as traditional convex optimization algorithm and learning-based solutions, cannot be directly utilized to solve the formulated optimization problem $\mathcal{P}1$ since USVs task execution mode selection, UAVs hovering coordinates and RIS phase shift vector are closely coupled. The proposed heuristic solution takes both the channel link quality between TBS and USVs and resource allocation of RIS-assisted UAV MEC network into consideration. In this regard, we first analyze UAVs flight route selection to promise that all USVs can be served by the most appropriate UAV. Then, USVs task execution mode selection is well studied to promise that each USV can dynamically determine task execution mode according to task data size and link quality to realize low latency data transmission. Afterward, we jointly consider RIS phase shift vector design and UAVs hovering coordinates, where we first obtain the feasible RIS phase shift vector via mathematical analysis and then obtain feasible UAVs hovering coordinates using the proposed ADMM-SCA-based algorithm. In this way,

one can obtain the feasible solution to $\mathcal{P}1$ efficiently.

V. Conclusion and Future Work

In this paper, an RIS-assisted UAV-enabled wireless inland ship MEC network architecture to support low-latency USVs data computation with time window is proposed. Aiming to enhance the QoS performance, the task processing time minimization problem is formulated by jointly considering UAV flight route selection, USVs execution mode selection, UAVs hovering coordinates and RIS phase shift vector design. To tackle the formulated challenging problem, a heuristic solution is proposed, where we first decouple the origin problem into three sub-problems, e.g., UAVs flight route selection optimization subproblem, USVs execution mode selection optimization subproblem and the joint UAVs hovering coordinates and RIS phase shift vector optimization subproblem, which are solved by the proposed EDA algorithm, ELRM algorithm and ADMM-SCA-based algorithm, respectively. In this way, the formulated challenging problem can be efficiently solved. Numerical results verify the effectiveness of the proposed solution compared with various selected advanced algorithms regarding task processing time. Also, the performance of the proposed solution in terms of typical UAVs caching capability and UAVs numbers has been investigated.

One of the most important issues for RIS-assisted UAV MEC networks is data safety, where wireless channels are observable to potential eavesdroppers. One widely used method to handle this scenario is to utilize redundancy in convolutional coding; however, this approach suffers a heavy communication load. Future research direction can be focused on reducing secrecy to avoid eavesdropping and maximize data transmission achievable rate via deep learning based and deep reinforcement learning algorithms [38-39]. Another promising research topic is artificial intelligence based secure communication for RIS-assisted UAV MEC networks. The existing solutions, such as reinforcement learning, Dinkelbach-guided DRL and so forth, can achieve better network security performance via jointly optimizing mobile devices offloading decisions and network resource allocation; however, most existing approaches cannot take UAV trajectory and RIS phase shift vector design into account. Besides, to release the ever-increasing heavy overload of USVs, a bi-objective optimization problem for USVs tasks execution energy consumption and overall tasks processing latency for RIS-assisted UAV MEC network worth further efforts since jointly taking both two performance parameters into account are usually complex and contradictory; very few works have been studied this emerging topic since the optimization process to solve this type of research problem is extremely challenging [40-41].

References

[1] G. Shao, Y. Ma, R. Malekian, X. Yan and Z. Li, "A Novel Cooperative Platform Design for Coupled USV–UAV Systems," IEEE Transactions on Industrial Informatics, vol. 15, no. 9, pp. 4913-4922, 2019.

[2] T. Yang, Y. Guo, Y. Zhou and S. Wei, "Joint Communication and Control for Small Underactuated USV Based on Mobile Computing Technology," IEEE Access, vol. 7, pp. 160610-160622, 2019.

[3] Y. Liao, X. Chen, S. Xia, Q. Ai and Q. Liu, "Energy Minimization for UAV Swarm-Enabled Wireless Inland Ship MEC Network With Time Windows," IEEE Transactions on Green Communications and Networking, vol. 7, no. 2, pp. 594-608, 2023.

[4] Q. Ai, X. Qiao, Y. Liao and Q. Yu, "Joint Optimization of USVs Communication and Computation Resource in IRS-Aided Wireless Inland Ship MEC Networks," IEEE Transactions on Green Communications and Networking, vol. 6, no. 2, pp. 1023-1036, 2022.

[5] Y. Liao, L. Lin, Y. Song and N. Xu, "Joint Communication-Caching-Computing Resource Allocation for Bidirectional Data Computation in IRS-Assisted Hybrid UAV-Terrestrial Network," Chinese Journal of Electronics, early access, 2023. doi: 10.23919/cje.2023.00.089, 2023.

[6] L. Zhang, Y. Sun, Z. Chen and S. Roy, "Communications-Caching-Computing Resource Allocation for Bidirectional Data Computation in Mobile Edge Networks," IEEE Transactions on Communications, vol. 69, no. 3, pp. 1496-1509, 2021.

[7] M. Asim, M. ELAffendi and A. El-Latif, "Multi-IRS and Multi-UAV-Assisted MEC System for 5G/6G Networks: Efficient Joint Trajectory Optimization and Passive Beamforming Framework," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 4, pp. 4553-4564, 2023.

[8] F. Spinelli and V. Mancuso, "Toward Enabled Industrial Verticals in 5G: A Survey on MEC-Based Approaches to Provisioning and Flexibility," IEEE Communications Surveys & Tutorials, vol. 23, no. 1, pp. 596-630, 2021.

[9] Y. Liao *et al.,* "Energy Minimization of Inland Waterway USVs for IRS-Assisted Hybrid UAV-Terrestrial MEC Network," IEEE Transactions on Vehicular Technology, vol. 73, no. 3, pp. 4121-4135, 2024.

[10] X. Qin *et al.,* "Joint Optimization of Resource Allocation, Phase Shift, and UAV Trajectory for Energy-Efficient RIS-Assisted UAV-Enabled MEC Systems," IEEE Transactions on Green Communications and Networking, vol. 7, no. 4, pp. 1778-1792, 2023.

[11] Z. Zhai, X. Dai, B. Duo, X. Wang and X. Yuan, "Energy-Efficient UAV-Mounted RIS Assisted Mobile Edge Computing," IEEE Wireless Communications Letters, vol. 11, no. 12, pp. 2507-2511, 2022.

[12] S. M. Asiful Huda and S. Moh, "Survey on Computation Offloading in UAV-Enabled Mobile Edge Computing," Journal of Network and Computer Applications, vol. 201, article number 103341, 2022.

[13] Z. Lin, X. Chen and P. Chen, "Energy Harvesting Space-air-sea Integrated Networks for MEC-enabled Maritime Internet of Things," China Communications, vol. 19, no. 9, pp. 47-57, 2022.

[14] Q. V. Pham, M. Zeng, T. Huynh-The, Z. Han and W. J. Hwang, "Aerial Access Networks for Federated Learning: Applications and Challenges," IEEE Network, vol. 36, no. 3, pp. 159-166, 2022.

[15] F. Naeem, G. Kaddoum, S. Khan, K. S. Khan and N. Adam, "IRS-Empowered 6G Networks: Deployment Strategies, Performance Optimization, and Future Research Directions," IEEE Access, vol. 10, pp. 118676-118696, 2022.

[16] T. Truong, V. Tuong, N. Dao and S. Cho, "FlyReflect: Joint Flying IRS Trajectory and Phase Shift Design Using Deep Reinforcement Learning," IEEE Internet of Things Journal, vol. 10, no. 5, pp. 4605-4620, 2023.

[17] H. Nguyen-Kha, H. V. Nguyen, M. T. P. Le and O. S. Shin, "Joint UAV Placement and IRS Phase Shift Optimization in Downlink Networks," IEEE Access, vol. 10, pp. 111221-111231, 2022.

[18] J. Liu *et al.,* "Efficient Dependent Task Offloading for Multiple Applications in MEC-Cloud System," IEEE Transactions on Mobile Computing, vol. 22, no. 4, pp. 2147-2162, 2023.

[19] C. Liu, K. Li, J. Liang and K. Li, "COOPER-MATCH: Job Offloading with A Cooperative Game for Guaranteeing Strict Deadlines in MEC," IEEE Transactions on Mobile Computing, early access, 2023, doi: 10.1109/TMC.2019.2921713.

[20] X. Chen, Y. Liao, Q. Ai and K. Zhang, "Joint Task Offloading and Resource Allocation for MEC Networks Considering UAV Trajectory," in 17th International Conference on Mobility, Sensing and Networking, Exeter, United Kingdom, pp. 296-302, 2021.

[21] Y. Sun, L. Zhang, Z. Chen and S. Roy, "Communications-Caching-Computing Tradeoff Analysis for Bidirectional Data Computation in Mobile Edge Networks," in IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), Victoria, BC, Canada, pp. 1-5, 2020.

[22] Y. Sun, L. Zhang, Z. Chen and S. Roy, "Communications-Caching-Computing Tradeoff Analysis for Bidirectional Data Computation in Mobile Edge Networks," in IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), Victoria, BC, Canada, pp. 1-5, 2020.

[23] L. Dong, Z. Liu, F. Jiang and K. Wang, "Joint Optimization of Deployment and Trajectory in UAV and IRS-Assisted IoT Data Collection System," IEEE Internet of Things Journal, vol. 9, no. 21, pp. 21583-21593, 2022.

[24] Q. Tao, S. Zhang, C. Zhong, W. Xu, H. Lin and Z. Zhang, "Weighted Sum-Rate of Intelligent Reflecting Surface Aided Multiuser Downlink Transmission With Statistical CSI," IEEE Transactions on Wireless Communications, vol. 21, no. 7, pp. 4925-4937, 2022.
[25] Y. Liao, X. Chen, J. Liu, Y. Han, N. Xu and Z. Yuan, "Cooperative UAV-USV MEC Platform for Wireless Inland Waterway Communications," IEEE Transactions on Consumer Electronics, 2023, early access, doi: 10.1109/TCE.2023.3327401.
[26] C. Zhong *et al.,* "Deep Reinforcement Learning-Based Optimization for IRS-Assisted Cognitive Radio Systems," IEEE Transactions on Communications, vol. 70, no. 6, pp. 3849-3864, 2022.
[27] C. Huang, G. Chen and K. K. Wong, "Multi-Agent Reinforcement Learning-Based Buffer-Aided Relay Selection in IRS-Assisted Secure Cooperative Networks," IEEE Transactions on Information Forensics and Security, vol. 16, pp. 4101-4112, 2021.
[28] Y. Gonczarowski, N. Nisan, R. Ostrovsky and W. Rosenbaum, "A Stable Marriage Requires Communication," Games and Economic Behavior, vol. 118, pp. 626-647, 2019.
[29] Y. Liao, Y. Song, L. Liu and Y. Han, "Joint Deployment and Task Scheduling in IRS-assisted Wireless Inland Ship MEC Network," in IEEE Vehicular Technology Conference, Florence, Italy, pp. 1-6, 2023.
[30] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming," version 2.1, 2014.
[31] A. Liu, V. K. N. Lau and B. Kananian, "Stochastic Successive Convex Approximation for Non-Convex Constrained Stochastic Optimization," IEEE Transactions on Signal Processing, vol. 67, no. 16, pp. 4189-4203, 2019.
[32] D. Ma, M. Ding and M. Hassan, "Enhancing Cellular Communications for UAVs via Intelligent Reflective Surface," in IEEE Wireless Communications and Networking Conference, Seoul, South Korea, pp. 1-6, 2020.
[33] J. Yu, B. Zhang, W. Chen, J. Zhang, "4G TD-LTE Radio Coverage Model Optimization Design Under Complex Inland River Environment," in International Conference on Transportation Information and Safety, Wuhan, China, pp. 442-446, 2015.
[34] Y. Liao *et al.*, "Energy Minimization for IRS-assisted UAV-empowered Wireless Communications," in 18th International Conference on Mobility, Sensing and Networking, Guangzhou, China, pp. 1001-1006, 2022.
[35] S. Singh and D. Kim, "Joint Optimization of Computation Offloading and Resource Allocation in C-RAN With Mobile Edge Computing Using Evolutionary Algorithms," IEEE Access, vol. 11, pp. 112693-112705, 2023.
[36] J. Liang, H. Xing, F. Wang and V. K. N. Lau, "Joint Task Offloading and Cache Placement for Energy-Efficient Mobile Edge Computing Systems," IEEE Wireless Communications Letters, vol. 12, no. 4, pp. 694-698, 2023.
[37] Y. Nie, J. Zhao, F. Gao and F. R. Yu, "Semi-Distributed Resource Management in UAV-Aided MEC Systems: A Multi-Agent Federated Reinforcement Learning Approach," IEEE Transactions on Vehicular Technology, vol. 70, no. 12, pp. 13162-13173, 2021.
[38] W. Song, S. Rajak, S. Dang, R. Liu, J. Li and S. Chinnadurai, "Deep Learning Enabled IRS for 6G Intelligent Transportation Systems: A Comprehensive Study," IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 11, pp. 12973-12990, 2023.
[39] H. Yang *et al.,* "Deep Reinforcement Learning-Based Intelligent Reflecting Surface for Secure Wireless Communications," IEEE Transactions on Wireless Communications, vol. 20, no. 1, pp. 375-388, 2021.
[40] Z. Li, N. Zhu, D. Wu, H. Wang and R. Wang, "Energy-Efficient Mobile Edge Computing Under Delay Constraints," IEEE Transactions on Green Communications and Networking, vol. 6, no. 2, pp. 776-786, 2022.
[41] M. M. Adam, L. Zhao, K. Wang and Z. Han, "Beyond 5G Networks: Integration of Communication, Computing, Caching, and Control," China Communications, vol. 20, no. 7, pp. 137-174, 2023.

Yangzhe Liao received his BS degree in Measurement and Control Technology from Northeastern University, China in 2013 and PhD degree from the University of Warwick, UK, in 2017. Currently, he is an associate professor at the School of Information Engineering, Wuhan University of Technology, China. His research interests include mobile edge computing and mobile computing.

Yuanyan Song obtained his BS degree from the Wuhan University of Technology, China in 2022. Currently, he is pursuing her master's degree at the School of Information Engineering, Wuhan University of Technology, China. His research interests including mobile edge computing and networking.

Shuang Xia obtained her BS degree from the Wuhan University of Technology in 2021. Currently, she is pursuing her master's degree at the School of Information Engineering, Wuhan University of Technology. Her research interests mainly focus on wireless resource allocation and and networking.

Yi Han received the BEng. degree from the International School of Software, Wuhan University, China, in 2010, and the MS degree in telecommunication from Dublin City University, Ireland in 2011. He then obtained the PhD degree from the Performance Engineering Laboratory, University College Dublin, Ireland. He is currently an associate professor with the School of Information Engineering, Wuhan University of Technology, China. His research interests include QoE assessment and performance-aware adaptive multimedia delivery.

Ning Xu received his Ph.D. degree in Electronic Science and Technology from the University of Electronic Science and Technology of China, Chengdu, in 2003. Later, he was a postdoctoral fellow with Tsinghua University, Beijing, from 2003 to 2005. Currently, he is a professor at the School of Information Engineering of Wuhan University of Technology, Wuhan. Dr. Xu's research interests include computer aided design of VLSI circuits and systems, image processing, big data analysis, and artificial intelligence.

Xiaojun Zhai is currently a Reader in the Embedded Intelligent Systems Laboratory at the University of Essex. He has authored/co-authored over 140 scientific papers in international journals and conference proceedings. His research interests mainly include the design and implementation of the digital image and signal processing algorithms, custom computing using FPGAs, embedded systems, and hardware/software co-design.

Zhenhui Yuan is an Assistant Professor at the University of Warwick, UK. He received his BEng degree in Software Engineering at Wuhan University, China, in 2008 and PhD in Electronic Engineering at Dublin City University, Ireland, in 2012. He is primarily interested in the development of communication systems for robots and vehicles. He was a postdoctoral researcher at Ericsson and Enterprise Ireland and a visiting scholar at UCLA, U.S. He was also a 5G Senior Researcher at Huawei (Shanghai). He co-founded RobSense (Hangzhou) Technology Co.Ltd which provides industrial IoT solutions with robotics. He was an Associate Professor at Hangzhou Dianzi University, China, and a Senior Lecturer at Northumbria University, UK. He won the best paper awards at IEEE ICCRE 2016 and IEEE BMSB 2014. He served as the Lead Guest Editor in IEEE Network and IEEE Internet of Things Journal, and an Associate Editor at IEEE ACCESS, IEEE Transactions on Consumer Electronics and Journal of Pervasive Computing & Communications. He is the founding chair of VeSUS (6G-empowered Robotic Vehicles for Sustainable Development) workshop. He is a voting member of IEEE P1954 on UAV communications.