Multi-Tier Hybrid Offloading for Computation-Aware IoT Applications in Civil Aircraft-Augmented SAGIN

Qian Chen, *Graduate Student Member, IEEE*, Weixiao Meng, *Senior Member, IEEE*, Tony Q. S. Quek, *Fellow, IEEE*, and Shuyi Chen, *Member, IEEE*

***Abstract*— Satellites and civil aircrafts (CAs) with computing ability are valuable access platforms, making it possible for Internet of Things (IoT) devices to offload their computation-intensive tasks in remote areas without network infrastructures. Unlike existing works mainly focused on the static scenarios or the interaction between any two types of local, edge and cloud nodes, we propose an innovative multi-tier hybrid parallel computation architecture in CA-augmented space-air-ground integrated networks (CAA-SAGIN). Specifically, devices perform local computing, CAs and satellites act as edge servers, and ground stations of satellite networks operate cloud computing. Aiming to minimize the weighted sum of end-to-end (E2E) delay and energy consumption, we formulate a partial computation offloading problem by jointly considering access strategy, transmit power, computing resource allocation, offloading ratio and delay tolerance. The platform selection exists both within and between layers, and there are inner- and inter-coupling relationships between communication and computing resources. The issue is solved by the proposed multi-tier partial task offloading (MPTO) algorithm. The original problem is firstly decomposed into primal and master subproblems by generalized benders decomposition (GBD) method, and parallel successive convex approximation (SCA) theory is utilized to transform the multi-variable NP-hard master problem into a convex one. Simulation results demonstrate the convergence and optimality of the MPTO algorithm and the advantages of this multi-tier hybrid computation offloading system. Also, the optimal tradeoff between E2E delay and energy consumption can be achieved by the MPTO algorithm.**

***Index Terms*— Internet of Things (IoT), multi-tier hybrid offloading, resource management, space-air-ground integrated networks (SAGIN), successive convex approximation (SCA).**

Manuscript received 16 May 2022; revised 1 September 2022; accepted 25 October 2022. Date of publication 9 December 2022; date of current version 19 January 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62271168; in part by the National Key Research and Development Program of China under Grant 2020YFE0205800; in part by the Young Elite Scientist Sponsorship Program by CAST under Grant YESS20210339; in part by the Fellowship of China Postdoctoral Science Foundation under Grant 2021TQ0092; in part by the Heilongjiang Postdoctoral Fund LBH-Z21001; and in part by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research and Development Programme. *(Corresponding author: Weixiao Meng.)*

Qian Chen, Weixiao Meng, and Shuyi Chen are with the Communications Research Center, Harbin Institute of Technology, Harbin 150001, China (e-mail: joycecq@163.com; wxmeng@hit.edu.cn; chenshuyitina@gmail.com).

Tony Q. S. Quek is with the Information System Technology and Design, Singapore University of Technology and Design, Singapore 487372 (e-mail: tonyquek@sutd.edu.sg).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2022.3227031.

Digital Object Identifier 10.1109/JSAC.2022.3227031

I. Introduction

WITHIN the era of 5G adoption proliferating, connectivity plays an essential role in how people live and communicate. It is estimated that total Internet of Things (IoT) connections will grow to 23.3 billion in 2025 [1]. Such massive devices promote enormous computation-intensive applications, such as smart cities [2], virtual reality [3], real-time video analytics [4], and environment monitoring [5]. Due to the limited computing capability of devices, it is challenging to execute all the tasks locally within the given time.

Mobile edge computing (MEC) and cloud computing (CC) are prospective methods to solve this issue by offloading part or all the tasks from local to edge or cloud servers [6]. The edge servers in the middle tier are typically just "one hop" distance from the devices but are featured by limited computing resources. In contrast, the cloud servers in the remote tier are usually far away from the user but practically have infinite resources [7]. With the benefits of computation offloading, it is necessary to exploit the tradeoff between transmission and computing delay and between delay and energy consumption. Therefore, the bottleneck problem of MEC and CC is resource management to deal with the complex coupling relationship between communication and computing resources. There is sufficient research on MEC and CC for terrestrial networks (TN), aiming at minimizing end-to-end (E2E) delay [7], [8], [9], energy consumption [10], [11], [12], [13], or the combination of delay and energy consumption [14], [15].

However, computation offloading is still complicated in harsh areas like oceans and forests without cellular coverage. Fortunately, satellite and aerial communications are developing rapidly and have become vital supplementary components for the next-generation wireless networks. Low-earth orbit (LEO) satellites launched by SpaceX have several custom drivers to interface with their hardware [16]. These satellites with distributed computer systems can act as edge servers. Similar to Starlink, the constellation of Iridium, ORBCOMM, and Telesat also have on-board processing capability rather than just forwarding data transparently. Civil aircraft (CA), as another kind of access platform in aeronautical networks (AN), can enhance the current satellite-terrestrial networks in terms of coverage [17], capacity [18], and task scheduling [19]. This novel network can be called CA-augmented space-air-ground integrated networks (CAA-SAGIN) [20]. One of the trends of

connected CAs is enabling data processing and analysis in-cabin [21], making it possible for CAs to be edge servers. For simplicity, the edge servers (i.e., CAs and satellites) are collectively called sky access platforms (SAPs) in this paper. Meanwhile, the ground stations (GSs) of satellite networks (SN) connected to the core network can act as cloud servers and execute the tasks. In virtue of intra- and inter-layer communication links, the information interaction and the awareness of computing resources can be realized in the local area. In this local-edge-cloud computation-aware system, we desire to explore the computation offloading scheme in CAA-SAGIN.

Compared to MEC and CC in TN, the main challenges of the multi-tier computation offloading in CAA-SAGIN are listed as follows.

1) *Intermittent links incurred by dynamic topology:* Due to the dynamic of SAPs and the sparse distribution of their corresponding GSs, the visible SAPs are changing between different time slots (TSs). Also, the links between SAPs and cloud servers are intermittent.
2) *Communication and computing constraints:* The distance and the fading of user-to-SAP links vary constantly. Thus, it is necessary to adjust users' transmit power to meet their quality of experience (QoE) requirements, considering the devices are energy-constrained with limited transmitting capability. In addition, since a few SAPs serve multiple computation-intensive tasks, the computing constraints of SAPs and devices are required to be considered.
3) *Intra- and inter-layer association strategy:* The satellites moving regularly in high orbit can almost provide ubiquitous coverage for users while introducing more propagation delay. Although the CAs with lower flight height can provide service enhancement and reduce the propagation delay, their coverage has uncertainty. Thus, the platform type and specific SAP to access must be appropriately determined.
4) *Delay-energy tradeoff in multi-tier parallel processing:* In the local-edge-cloud computing framework, different computing platforms have their advantages. With the uneven communication and computation resources distribution in the CAA-SAGIN, it is difficult to balance the multi-dimensional limited resources to strike the performance tradeoff between E2E delay and energy consumption.

Motivated by these facts, we fully exploit all the computing platforms in CAA-SAGIN and construct a multi-tier hybrid parallel computation system in this paper. IoT devices execute local processing, satellites and airplanes act as edge servers, and GSs of SN operate CC. Considering the attributes of tasks, coupling of communication and computation, and resource constraints, this paper aims to answer four questions: 1) After the user generates a task, how to select the optimal platform to offload if there are multiple SAPs within its range? 2) How to adjust the user's transmit power while considering its transmitting capability and QoE requirements? 3) To satisfy the delay tolerance of the computation-intensive tasks, how to allocate the proportion of tasks at different platforms? 4) Given computing capability, how to allocate computational resources for each platform? To the best of our knowledge, this is the first paper to study the tradeoff between E2E delay and energy consumption in the local-edge-cloud hybrid offloading system in SAGIN. Overall, the contributions of this paper can be summarized as follows:

- We orchestrate all the platforms with computing ability in CAA-SAGIN and propose a multi-tier hybrid offloading computation-aware system with local, edge and cloud computing simultaneously. A universal resource management method for dynamic integrated networks is investigated under the parallel processing strategy.
- Considering the strict QoE of computation-intensive IoT applications and limited communication and computation resources, we formulate a task offloading optimization problem to achieve the performance tradeoff between delay and energy consumption. The multi-variable coupling problem is decomposed into two subproblems by the generalized benders decomposition (GBD) method. The first is a convex primal problem related to the users' transmit power, providing an upper bound of the original problem. The other is an NP-hard master problem related to user association strategy, task proportion and resource allocation, providing the lower bound of the problem.
- Based on the parallel successive convex approximation (SCA) theory, the non-convex master problem is transformed into a tractable convex one with the modified parallel SCA algorithm. The original optimization problem is solved by the proposed multi-tier partial task offloading (MPTO) algorithm. Simulations are provided to verify the convergence and optimality of the MPTO algorithm and the effectiveness of the multi-tier computation architecture. Also, the proposed MPTO algorithm can approach the optimal tradeoff between E2E delay and energy consumption.

The rest of this paper is organized as follows. Related works are elaborated in Section II. The system model is described in Section III, including the network model, transmission and computing models. Aiming to jointly minimize the weighted sum of E2E delay and energy consumption, we formulate a computation offloading optimization problem and decompose it into primal and master problems by the GBD approach in Section IV. Then, the original problem is solved by the proposed MPTO algorithm in Section V. Simulation results are discussed in Section VI, and conclusions are provided in Section VII.

II. Related Work

A. Resource Management in Edge and Cloud Computing

Many works have exploited the benefits of offloading local tasks to external servers in TN. These works can further be divided into three branches in terms of computation architecture: local-edge computing [10], [11], [12], [13], [14], edge-cloud computing [8], and local-edge-cloud computing [7], [9].

Feng et al. focused on the cooperation problem between MEC servers' ultra-reliable and low-latency communication (URLLC) transmission in vehicular networks for the first time, aiming to minimize the energy consumption of edge

TABLE I
SUMMARY OF RELATED WORKS ON SAGIN-ENABLED COMPUTATION OFFLOADING

SAGIN-Enabled Computation Offloading Works			
	UAV-Terrestrial	Satellite-Terrestrial	Satellite-UAV-Terrestrial
Network Architecture	[23], [26], [30], [33], [38]	[27]–[29], [31], [32], [35]	[34], [36], [37]
	Local-Edge	Edge-Cloud	Local-Edge-Cloud
Computation Framework	[23], [26]–[28], [38]	[29]–[34]	[35]–[37]
	Binary Offloading	Partial Offloading	
Offloading Strategy	[26], [29], [31], [32], [35], [37], [38]	[23], [27], [28], [30], [33], [34], [36]	

servers and vehicles while satisfying the stability of task queuing [13]. Song et al. investigated the tradeoff between the task scheduling delay and the energy consumption of devices while considering the task precedence as a new constraint [14]. However, these two works adopted a binary offloading strategy, denoting that all the tasks are executed in the same place after determining the computing platform. On the contrary, partial offloading enables the missions to be partitioned, and each part can be processed at different platforms [22]. Due to parallel computing, partial offloading performs better than binary one on system performance. Meanwhile, resource management is more challenging under the partial offloading strategy because it answers which platform to access and how many tasks to be offloaded simultaneously.

The energy consumption minimization problems were formulated under partial offloading schemes [11], [12]. However, the computing resources allocated to different tasks were assumed to be the same fixed values [12], which neglected the tasks' heterogeneity and the constraints of the limited computing resources of local and edge servers. Also, the upper bound of task delay was not taken into consideration [8], [11], [12], [13], [14], which is not suitable for our investigated computation-intensive tasks.

To fully reap the benefits of each type of computing entity, Cardellini et al. formulated a generalized Nash equilibrium problem (GNEP) and analyzed the computation offloading in a local-edge-cloud computing scenario for the first time [7]. These non-cooperative devices can selfishly make offloading decisions in the decentralized system, aiming to minimize task response time. To study the advantages of combining MEC and CC, Ning et al. formulated the E2E delay minimization problems under both single user and multiuser situations, which can be solved by the branch and bound algorithm and the proposed iterative heuristic MEC resource allocation algorithm respectively [9]. However, the works on terrestrial three-tier computing remain at the stage of the binary offloading strategy.

B. SAGIN-Enabled Computation Offloading

The MEC techniques for TN cannot be applied for the scenario where terrestrial users increase sharply, or network facilities are insufficient [23]. Thus, implementing computation offloading in SAGIN is indispensable for computation-intensive tasks in remote areas [24], [25].

Existing works of SAGIN-enabled computation offloading can be categorized as shown in Table I. Hu et al. investigated the problem of computation efficiency maximization for UAV-enabled MEC systems while guaranteeing the QoE of users [26]. Under the partial offloading strategy, Hu et al. optimized the UAV trajectory, the offloading ratio, and user-association results to minimize the users' maximum delay [23]. Wang et al. proposed a game-theoretic computation offloading scheme in the satellite edge computing network to jointly minimize the average response time and energy consumption [27]. Song et al. investigated computation offloading from IoT devices to satellites and minimized the weighted-sum energy consumption of users when considering the delay constraint of tasks [28]. However, they set the CPU-cycle frequency of edge servers as a fixed value and failed to allocate it according to different attributes of the collected tasks.

In terms of edge-cloud computing offloading, Cui et al. studied the latency and energy optimization problem in satellite-IoT-enabled MEC system [29]. After decomposing the original problem, the two subproblems can be solved by the Lagrange multiplier method and Markov decision process (MDP), respectively. Duan et al. analyzed the optimal resource allocation strategy in multi-UAV systems by jointly optimizing task scheduling and computation resources [30]. Yu et al. proposed a SAGIN-enabled MEC to support the Internet of vehicles in remote areas, aiming at minimizing the task completion time and satellite resource usage [31]. Cao et al. extended this architecture by introducing software-defined networking (SDN) and network function virtualization (NFV), and proposed an improved algorithm considering delay, energy, resource utilization and security [32]. Yu et al. proposed a UAV-enabled MEC system where computing tasks can be assigned between UAV and nearby edge clouds [33]. Zhou et al. proposed an online offloading scheme and formulated a constrained MDP to minimize the long-term average system delay [34]. Nevertheless, these works failed to consider the local processing of IoT devices. Also, they neglected the delay tolerance of the tasks, which is an essential attribute of computation-intensive tasks.

As for the three-tier computation offloading works in SAGIN, Zhang et al. studied the joint computing and communication resource allocation problem to minimize the execution delay, where devices, satellites, and GSs perform local, edge, and cloud computing, respectively [35]. However, they discarded the benefits of parallel processing and the energy constraints of the devices. Liao et al. presented a learning-based queue-aware task offloading and resource allocation algorithm (QUARTER), aiming to minimize the average energy consumption [36]. However, the computing capability of satellites was missing, and the task delay tolerance was omitted. Cheng et al. also utilized the MDP and proposed a deep reinforcement learning-based approach to obtain the optimal offloading scheme under binary offloading strategy [37].

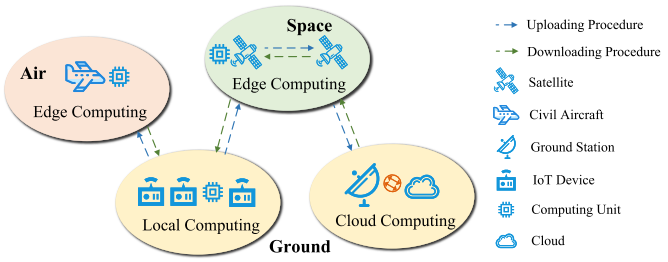


Fig. 1. Multi-tier hybrid offloading framework in CAA-SAGIN and illustration of task offloading.

The simulation results verified its convergence and efficiency. However, they also neglected the processing capability of satellites as edge servers and the dynamic allocation of local computing resources for different tasks.

All these three-tier works in SAGIN failed to discuss adjusting users' transmit power to satisfy their QoE requirements in different TSs. Moreover, they assumed that there is only one visible satellite, which can always connect with cloud servers directly. This assumption is unsuitable for the current mega-constellation scenarios and the practical situation in which the links between satellites and core networks are intermittent. Also, they neglected the offloading between edge and remote servers, and none of them discussed the tradeoff between E2E delay and energy consumption in SAGIN-enabled computation offloading systems.

III. System Model

A. Network Model

Fig. 1 illustrates the multi-tier hybrid offloading framework and the procedure of task offloading in CAA-SAGIN. The investigated time horizon is divided into T TSs with the same duration τ, and TSs are indexed by a set $t \in \mathcal{T} = \{1, \ldots, T\}$. The network topology is assumed to be invariant in each TS and may change between different TSs. There are U IoT devices (users) distributed on the ground in a remote area without cellular coverage, which are denoted as $u \in \mathcal{U} = \{1, \ldots, U\}$. Compared to SAPs, the IoT devices' speed is much lower. Thus, the relative motion between users and SAPs mainly relies on the movement of SAPs, and these devices can be regarded as quasi-static. They generate different computation-intensive applications intermittently, and the set of the generated tasks is denoted as $f \in \mathcal{F} = \{1, \ldots, F\}$. We use a tuple $\left\{\mathrm{st}_f, b_f, T_f^{\max}\right\}$ to characterize the task f, where st_f is the user which generates the task, b_f is the input number of packets, and $T_f^{\max}$ is the tolerable latency to complete the task. The delay requirements of these tasks are often strict. That is, the procedure of task scheduling is transient, while the variation of network topology is slow in comparison. Thus, the network topology is considered to be fixed within one task scheduling period [35]. For clarity, the important notations are listed in Table II.

Due to the limited computing capability of IoT devices, they need other platforms for parallel processing to guarantee the latency requirements. Let A and S denote the number of airplanes and satellites respectively, and the sets of airplanes and satellites are denoted as $a \in \mathcal{A} = \{1, \ldots, A\}$ and $s \in \mathcal{S} = \{1, \ldots, S\}$ respectively. For simplicity, let $i \in \mathcal{I} = \mathcal{A} \cup \mathcal{S}$ denote the set of all the SAPs. Since SN and AN operate at Ka- and V-band respectively, there is no inter-layer interference. When these SAPs fly above the given area, they will send signals by downlink transmission links on the pilot channel, which announce their position, communication and computing resource occupation situation, and other necessary information. Then, the IoT devices within these SAPs' coverage will send their computation requirements and capability to their visible SAPs by ground-to-air (G2A) and ground-to-space (G2S) links. The SAPs can change their collected information with each other by inter-satellite links (ISLs) and air-to-space (A2S) links, realizing the information interaction within and between layers and constructing a computation-aware system. In this paper, SAPs with computing units are the entities to execute the proposed optimization algorithms. Meanwhile, they can act as edge server platforms to process the computation-intensive missions onboard and then send the results back to users. Since the data size transmitted during information interaction is limited, the delay of the above procedure can be neglected.

After generating an application, the user chooses a CA or satellite as an edge server. To describe the user association strategy, we introduce a binary matrix $\mathbf{x} = \{x_f(u,i) | u \in \mathcal{U}, i \in \mathcal{I}, f \in \mathcal{F}\}$ in which $x_f(u,i) = 1$ indicates that user $u = \mathrm{st}_f$ accesses SAP i to complete task f's processing. Otherwise, $x_f(u,i) = 0$. Each task is assumed to offload to at most one SAP. Then, we have $\sum_{i \in \mathcal{I}} x_f(u,i) \leq 1, \ \forall f \in \mathcal{F}$. The multi-antenna SAPs can serve multiple users in one TS.

When the tasks are offloaded to SN, the accessed satellites with forwarding capability can also offload part of their collected data to GSs for CC. The set of the GSs is denoted as $g \in \mathcal{G} = \{1, \ldots, G\}$. With proper site selection of GSs, the distance between GSs and the core network is usually limited. Since GSs connect to the core network by high-speed wired links with light velocity, the transmission delay from GSs to the core network can be neglected. Thus, tasks arriving at GSs are equivalent to transmitting to the core network. It should be noticed that the processing capability of the core network is powerful enough.

Note that the CA gateways are hardly built in remote areas, and the communication range of CAs is smaller than that of satellites. Thus, the links between CA and CA gateway cannot be set up in this case. Since the relative velocity and flight direction of different CAs change rapidly and constantly between different TSs, the air-to-air (A2A) links are unstable as ISLs. It is also quite complex to transfer the packets to the CA gateways by A2S, ISLs and space-to-ground (S2G) link simultaneously. Therefore, when the tasks are offloaded to AN, CAs process their collected tasks at their computing units, and we do not consider the case that airplanes offload data to CA gateways directly or by other CAs with A2A links.

Users must also decide how many data packets to offload to the accessed SAP. Let $\boldsymbol{\rho} = \{\rho_f \in [0,1] | f \in \mathcal{F}\}$ denote the ratio of task f processed locally. Let R_p be the number

TABLE II
SUMMARY OF MAIN NOTATIONS

Notation	Definition
$\mathcal{S}, \mathcal{A}, \mathcal{I}, \mathcal{U}, \mathcal{G}$	The set of satellites $s \in \mathcal{S}$, CAs $a \in \mathcal{A}$, SAPs $i \in \mathcal{I}$, users $u \in \mathcal{U}$ and GSs $g \in \mathcal{G}$.
$\mathcal{T}, \tau$	The set of whole time horizon $t \in \mathcal{T}$, and the length of each TS.
$\mathcal{F}, \mathrm{st}_f, b_f, T_f^{\max}$	The set of generated tasks $f \in \mathcal{F}$. The user which generates f. Data amount and tolerable latency of f.
$W_{\mathrm{UL}}(u,i), \gamma_f(u,i), R_f(u,i), R_{\mathrm{th},f}$	The bandwidth, received SNR, achievable data rate, and threshold rate for uplink transmission from u to i.
$x_f(u,i)$	Binary variable, indicating whether u's task f is offloaded to SAP i.
α, R_p, ρ_f	The ratio of the output data size to input one, the number of bits per packet, and the ratio of task f processed locally.
$p_{\mathrm{ISL}}, R_{\mathrm{ISL}}, n_f^{\mathrm{hop}}$	The transmit power of satellites by ISLs, the achievable data rate of ISLs, and the number of hops by ISLs.
$\beta_f(s,g)$	Binary variable, indicating the visibility of GS g for satellite s.
$p_f^{\mathrm{TX}}(u,i), p^{\mathrm{RX}}(u,i), P_{\max}(u,i)$	Transmit and receive power from u to i. Maximal transmit power of u when accessing i.
$y_f(u,i), y_f^{\mathrm{EC}}(i), y_f^{\mathrm{CC}}(g)$	The number of packets offloaded from u to i, processed at SAP i for MEC, and at GS g for CC.
$T_f(u,i), T_f(s,s_{\max}), T_f(s_{\max},g)$	Uplink transmission time from u to i, total forwarding delay by ISLs during uploading procedure, transmission delay from the visible satellite $s_{\max}$ to g
$T_f^{\mathrm{AN,pro}}(u,a), T_f^{\mathrm{SN,pro,1}}(u,s), T_f^{\mathrm{SN,pro,2}}(u,s)$	Propagation delay when accessing AN by a, accessing SN by s for MEC, accessing SN and sending packets to GS for CC by s.
$T_f^{\mathrm{LC}}(u), T_f^{\mathrm{Off}}(u), T_f^{\mathrm{total}}(u)$	Local execution latency, offloading delay, and total delay of u's task f
$E_f(u,i), E_f^{\mathrm{LC}}(u), E_f^{\mathrm{EC}}(i), E_f^{\mathrm{CC}}(g), E_f^{\mathrm{total}}(u)$	Energy consumption of uploading data from u to i, local computing at u, MEC at i, CC at g, and total energy consumption of executing task f.
$\omega_u, \omega_i, \kappa_u, \kappa_i$	The required number of CPU cycles to compute one-bit input data of u and i. The constant depending on the chip architecture of u and i
$n_f^{\mathrm{LC}}(u), n_f^{\mathrm{EC}}(i), N_{\max}(u), N_{\max}(i)$	The computation capability of u and i allocated for task f. The maximal computing capacity of u and i.
η, δ_T, δ_E	Balancing factor defining the relative weight of E2E delay and energy consumption. Normalized factors of delay and energy consumption.
$\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\phi}, Q, M$	Lagrange multiplier vector when primal problem is feasible and infeasible. The variables of master problem. Auxiliary variable for infeasible case and for master problem.
$r_f, r_f(u,s), r_{f,1}(u,s), r_{f,2}(u,s)$	Auxiliary variables to transform $\mathcal{P}0$.
$\dot{U}_f(u,i), \dot{E}_f^{\mathrm{LC}}(u), \dot{y}_f^{\mathrm{CC}}(g)$	The convex approximation functions of the non-convex terms in master problem.
l, l_1, l_2, k	Iteration number of the primal problem, iteration number when primal problem is feasible and infeasible. Iteration number of master problem.
UBD, LBD	The upper bound and lower bound of $\mathcal{P}1$.

of bits in each packet. Thus, the bits of data amount for local computing is $\rho_f b_f R_p$, and the user offloads $y_f(u,i) = (1-\rho_f) b_f$ packets to its associated SAP. Define $\mathbf{y} = \left\{ y_f^{\mathrm{EC}}(s) \middle| s \in \mathcal{S}, f \in \mathcal{F} \right\}$ as the matrix of the number of packets processed at satellite s for MEC. Let $y_f^{\mathrm{CC}}(g)$ denote the number of packets offloaded to GS g for CC, then $y_f^{\mathrm{EC}}(s) + y_f^{\mathrm{CC}}(g) = y_f(u,s)$ holds. When user accesses CA a, $y_f^{\mathrm{EC}}(a) = y_f(u,a)$ holds.

B. Transmission During Uploading Procedure

The procedures of users offloading their tasks to different SAPs are shown in Fig. 2.

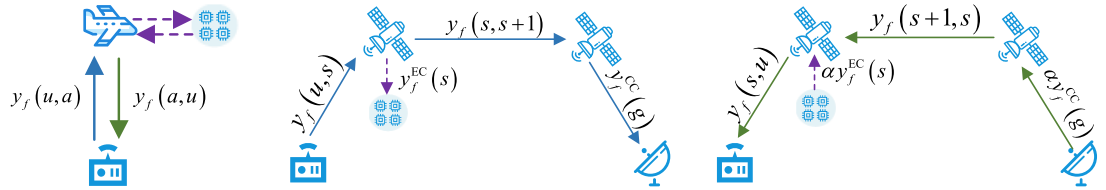
1) Uplink From User to SAP: By adopting multiplex methods like four-color multiplex, the nearby homogeneous SAPs can occupy different spectrums to reduce the inter-cell interference below the threshold value. Orthogonal multiple access is applied for frequency resources [19]. Since the number of active users is limited in each cell and broadband communications are feasible [39], the orthogonal frequency resources are relatively sufficient for these computation-intensive packets. Thus, the intra-cell interference can also be neglected. Based on these facts, we discuss the signal-to-noise ratio (SNR) in this paper.

Denote $\mathbf{p} = \left\{ p_f^{\mathrm{TX}}(u,i) \middle| u \in \mathcal{U}, i \in \mathcal{I}, f \in \mathcal{F} \right\}$ as a matrix including the transmit power from user u to SAP i when transmitting f. For the uplink transmission from user u to SAP i, the received SNR at SAP i can be expressed as

$$\gamma_f(u,i) = \frac{p_f^{\mathrm{TX}}(u,i) G_T(u) G_R(i) \left(\frac{\lambda(u,i)}{4\pi d_f(u,i)}\right)^2 L_a(u,i) |h_f(u,i)|^2}{n_0 W_{\mathrm{UL}}(u,i)}, \tag{1}$$

where $G_T(u)$ and $G_R(i)$ are transmit antenna gain of user u and receiving antenna gain of SAP i. $\left(\frac{\lambda(u,i)}{4\pi d_f(u,i)}\right)^2$ denotes free-space path loss, where $\lambda(u,i)$ and $d_f(u,i)$ are the wavelength of transmitted signal and distance between user u and SAP i. $L_a(u,i)$ represents the additional loss caused by atmosphere and environment. $h_f(u,i)$ is the Rician fading coefficient of link between user u and SAP i. n_0 is the noise power spectral density and $W_{\mathrm{UL}}(u,i)$ is the allocated bandwidth for uplink between user u and SAP i. According to the Shannon-Hartley theorem, the achievable data rate of uplink transmission from user u to SAP i can be given by

$$R_f(u,i) = W_{\mathrm{UL}}(u,i) \cdot \log_2\left[1 + \gamma_f(u,i)\right]. \tag{2}$$



(a) Two-way procedure when accessing airplane. (b) Uploading procedure when accessing satellite. (c) Downloading procedure when accessing satellite.

Fig. 2. Illustration of the offloading procedure of one task in SAP networks, where solid lines denote data transmission, and dash lines denote the tasks processed at the nodes.

Since $h_f(u,i)$ is a Rician fading coefficient, $|h_f(u,i)|^2$ follows non-central chi-square (χ^2) distribution. Let $K_f(u,i)$ be the angle-based Rician factor of link between user u and SAP i [40]. By setting the threshold rate for uplink transmission as $R_{\mathrm{th},f}$, the outage probability can be obtained by

$$\begin{aligned}&\mathbb{P}\left(R_f(u,i)<R_{\mathrm{th},f}\right)\\&=\mathbb{P}\left(|h_f(u,i)|^2<\frac{R_{f,0}(u,i)}{\left(\frac{\lambda(u,i)}{4\pi d_f(u,i)}\right)^2}\right)\\&\underline{(a)}\ 1-Q_1\left(\sqrt{2K_f(u,i)},\right.\\&\qquad\left.\frac{4\pi d_f(u,i)\sqrt{2\left(K_f(u,i)+1\right)R_{f,0}(u,i)}}{\lambda(u,i)}\right),\end{aligned}\tag{3}$$

where $R_{f,0}(u,i)=\frac{n_0 W_{\mathrm{UL}}(u,i)\left(2^{R_{\mathrm{th},f}/W_{\mathrm{UL}}(u,i)}-1\right)}{p_f^{\mathrm{TX}}(u,i)G_T(u)G_R(i)L_a(u,i)}$. Step (a) is obtained by the cumulative distribution function (CDF) of $|h_f(u,i)|^2$ as follows,

$$\begin{aligned}&F_{|h|^2}(z)\\&=\begin{cases}1-Q_1\left(\sqrt{2K_f(u,i)},\sqrt{2\left(K_f(u,i)+1\right)z}\right), & z>0,\\0, & \text{otherwise},\end{cases}\end{aligned}\tag{4}$$

where $Q_1(a,b)=\int_b^\infty x\mathrm{e}^{-\frac{a^2+x^2}{2}}I_0(ax)\mathrm{d}x$ is Marcum Q-function and $I_0(x)=\sum_{k=0}^{\infty}\left(\frac{x^k}{2^k k!}\right)^2$ is 0-th order modified Bessel function of the first kind.

Then, the uplink transmission time from u to i is given by

$$T_f(u,i)=\frac{\left(1-\rho_f\right)b_f R_p}{R_f(u,i)},\tag{5}$$

and the total energy consumption of uplink transmission is

$$E_f(u,i)=\left(p_f^{\mathrm{TX}}(u,i)+p^{\mathrm{RX}}(u,i)\right)T_f(u,i),\tag{6}$$

where $p^{\mathrm{RX}}(u,i)$ is the receiving power of SAP i. Note that $p_f^{\mathrm{TX}}(u,i)$ is a random variable which should be adjusted according to the type of SAP and the distance between user and SAP, while $p^{\mathrm{RX}}(u,i)$ is often a fixed value.

2) Data Forwarding Between Satellites: When the task is offloaded satellite s which is not within the range of the GS, s needs to forward its data to other satellites by ISLs until one satellite can see the GS. Assume that the traffic goes from the smallest number of satellites to the largest one during the uploading procedure in SN. That is, satellite $s=s_{\min}$ collects data from a user, satellite $s_{\max}$ transmits the data to GS, and other satellites from $s+1$ to $s_{\max}-1$ are used for data forwarding. Specifically, if s is visible to any GS, $s=s_{\max}$ holds. As shown in Fig. 2(b), the following expressions hold in the SN according to flow conservation,

$$\begin{aligned}&\left(1-\rho_f\right)b_f R_p\\&=y_f^{\mathrm{EC}}(s)+\left(1-\beta_f(s,g)\right)y_f(s,s+1)\\&\quad+\beta_f(s,g)\,y_f^{\mathrm{CC}}(g),\quad\forall s\in\mathcal{S}:x_f(u,s)=1,\end{aligned}\tag{7}$$

and

$$\begin{aligned}&y_f(s-1,s)\\&=\left(1-\beta_f(s,g)\right)y_f(s,s+1)\\&\quad+\beta_f(s,g)\,y_f^{\mathrm{CC}}(g),\quad\forall s\in\mathcal{S}:x_f(u,s)=0,\end{aligned}\tag{8}$$

where $\beta_f(s,g)\in\{0,1\}$ denotes the visibility of g for s. If satellite s is in the range of GS g, then $\beta_f(s,g)=1$. Otherwise, $\beta_f(s,g)=0$. $y_f(s,s+1)$ is the transmitted packets of task f from satellite s to its adjacent satellite $s+1$ during the uploading procedure.

Let p_{ISL} be the transmit and receive power of satellites when delivering data by ISLs, and R_{ISL} be the achievable data rate of ISL, which are fixed values. Then, the total forwarding delay by ISLs during uploading procedure can be given by $T_f(s,s_{\max})=\frac{n_f^{\mathrm{hop}}y_f^{\mathrm{CC}}(g)R_p}{R_{\mathrm{ISL}}}$, where n_f^{hop} is the number of hops which traffic f experiences from s to $s_{\max}$. The total energy consumption of transmission by ISLs during uploading procedure is $E_f(s,s_{\max})=2p_{\mathrm{ISL}}T_f(s,s_{\max})$. Since the propagation delay is severe in SN, we adopt the shortest path routing method in this paper [19], which is most likely to approach the optimal method.

3) Downlink From Satellite to Ground Station: Let R_f^{S2G} denote the achievable rate of the link between satellite s and GS g when transmitting f. Then, the transmission delay from satellite s to GS g can be given by $T_f(s,g)=\frac{y_f^{\mathrm{CC}}(g)R_p}{R_f^{\mathrm{S2G}}}$, and the corresponding energy consumption is $E_f(s,g)=\left(p^{\mathrm{TX}}(s,g)+p^{\mathrm{RX}}(s,g)\right)T_f(s,g)$.

C. Transmission During Downloading Procedure

After processing by other platforms, the computation result must be transmitted back to IoT devices. Considering that the data size after processing is smaller than that of input, a proportion factor α is introduced to represent the ratio of output data size to input one.

Since satellites and GSs can compute part of the tasks, they all have to transmit their processed results back to users. When GS g transmits its processed task to satellite s, $y_f(g,s)=\alpha y_f^{\mathrm{CC}}(g)$ holds. Since the network topology is assumed to

be unchanged during each task scheduling, the routes by ISLs during the downloading procedure can be considered the same as the uploading one. Thus, the transmission delay $T_f(g,s)$ and energy consumption $E_f(g,s)$ can be given by $T_f(g,s) = \frac{y_f(g,s)R_p}{R_f^{\mathrm{S2G}}} = \alpha T_f(s,g)$ and $\alpha E_f(s,g)$. The total forwarding delay between satellites during downloading procedure can be obtained by $T_f(s_{\max},s) = \frac{n_f^{\mathrm{hop}}\alpha y_f^{\mathrm{CC}}(g)R_p}{R_{\mathrm{ISL}}} = \alpha T_f(s,s_{\max})$ and the energy consumption is $E_f(s_{\max},s) = 2p_{\mathrm{ISL}}T_f(s_{\max},s) = \alpha E_f(s,s_{\max})$.

The transmission latency from SAP i to user u can be calculated by $T_f(i,u) = \frac{y_f(i,u)R_p}{R_f(i,u)}$, where $y_f(i,u) = \alpha y_f(u,i)$, and the energy consumption is $E_f(i,u) = \left(p^{\mathrm{TX}}(i,u) + p^{\mathrm{RX}}(u,i)\right)T_f(i,u)$.

D. Computing Model

1) Local Computing at IoT Devices: Let $n_f^{\mathrm{LC}}(u)$ denote the user u's allocated computation capability (i.e., CPU cycles/s) for task f. The required number of CPU cycles to compute one-bit input data of u is denoted by ω_u. Then, the local execution latency of u when computing f can be expressed as

$$T_f^{\mathrm{LC}}(u) = \frac{\omega_u \rho_f b_f R_p}{n_f^{\mathrm{LC}}(u)}, \tag{9}$$

and the corresponding energy consumption can be given by

$$E_f^{\mathrm{LC}}(u) = \kappa_u\left(n_f^{\mathrm{LC}}(u)\right)^3 T_f^{\mathrm{LC}} = \kappa_u\omega_u\rho_f b_f R_p\left(n_f^{\mathrm{LC}}(u)\right)^2, \tag{10}$$

where κ_u is a coefficient related to the chip architecture of user u.

2) Edge Computing at SAPs: The computation latency of SAP i as edge server when computing task f can be given by

$$T_f^{\mathrm{EC}}(i) = \frac{\omega_i y_f^{\mathrm{EC}}(i) R_p}{n_f^{\mathrm{EC}}(i)}, \tag{11}$$

where ω_i is required number of CPU cycles to compute one-bit input data of SAP i, and $n_f^{\mathrm{EC}}(i)$ is the computing capability of i allocated for task f. Then, the energy consumption of SAP i for MEC can be obtained by

$$E_f^{\mathrm{EC}}(i) = \kappa_i\left(n_f^{\mathrm{EC}}(i)\right)^3 T_f^{\mathrm{EC}}(i) = \kappa_i\omega_i y_f^{\mathrm{EC}}(i) R_p\left(n_f^{\mathrm{EC}}(i)\right)^2, \tag{12}$$

where κ_i is a constant depending on the chip architecture of SAP i.

3) Cloud Computing at Ground Stations: Since GS has multi-core high-speed CPUs and its computation capability is strong enough, the computing latency at GS can be neglected. The energy consumption of CC depends on the input data size, given by [30]

$$E_f^{\mathrm{CC}}(g) = A_0 \exp\left(A_1 y_f^{\mathrm{CC}}(g)\right), \tag{13}$$

where A_0 and A_1 are proportionality coefficients.

E. Parallel Processing

Parallel processing is enabled in CAA-SAGIN to improve computing efficiency. When a user accesses CA for partial offloading, the delay in AN for task f can be given by $T_f^{\mathrm{Off}}(u,a) = T_f(u,a) + T_f^{\mathrm{EC}}(a) + T_f(a,u) + T_f^{\mathrm{AN,pro}}(u,a)$, where $T_f^{\mathrm{AN,pro}}(u,a) = \frac{2d_f(u,a)}{c}$ is the propagation delay and c is the light speed.

When satellite is selected as the edge server, the latency of task f at the accessed satellite can be given by $T_f^{\mathrm{Off},1}(u,s) = T_f(u,s) + T_f^{\mathrm{EC}}(s) + T_f(s,u) + T_f^{\mathrm{SN,pro},1}(u,s)$, where $T_f^{\mathrm{SN,pro},1}(u,s) = \frac{2d_f(u,s)}{c}$. The latency of task f offloaded to GS g by satellite s can be expressed as $T_f^{\mathrm{Off},2}(u,s) = T_f(u,s) + T_f(s,s_{\max}) + T_f(s_{\max},g) + T_f(g,s_{\max}) + T_f(s_{\max},s) + T_f(s,u) + T_f^{\mathrm{SN,pro},2}(u,s)$, where $T_f^{\mathrm{SN,pro},2}(u,s) = \frac{2}{c}\left(d_f(u,s) + d_{f,\mathrm{ISL}} + d_f(s_{\max},g)\right)$ and $d_{f,\mathrm{ISL}}$ is the sum of the distance by ISLs. Then, the delay in SN for task f when accessing s can be obtained by $T_f^{\mathrm{Off}}(u,s) = \max\left\{T_f^{\mathrm{Off},1}(u,s), T_f^{\mathrm{Off},2}(u,s)\right\}$. Thus, for user u's task f, the delay of offloading to SAP is $T_f^{\mathrm{Off}}(u) = \sum_{i\in\mathcal{I}} x_f(u,i)\, T_f^{\mathrm{Off}}(u,i)$, and the total delay is given by

$$T_f^{\mathrm{total}}(u) = \max\left\{T_f^{\mathrm{LC}}(u), T_f^{\mathrm{Off}}(u)\right\}. \tag{14}$$

In addition, the energy consumption in AN is given by

$$E_f^{\mathrm{Off}}(u,a) = E_f(u,a) + E_f(a,u) + E_f^{\mathrm{EC}}(a). \tag{15}$$

The energy consumption in SN can be expressed as

$$E_f^{\mathrm{Off}}(u,s) = E_f^{\mathrm{Tran}}(u,s) + E_f^{\mathrm{EC}}(s) + E_f^{\mathrm{CC}}(g), \tag{16}$$

where $E_f^{\mathrm{Tran}}(u,s) = E_f(u,s) + E_f(s,s_{\max}) + E_f(s_{\max},g) + E_f(g,s_{\max}) + E_f(s_{\max},s) + E_f(s,u)$ denotes the transmission energy. Therefore, the total energy consumption of task f can be obtained by

$$E_f^{\mathrm{total}}(u) = E_f^{\mathrm{LC}}(u) + \sum_{i\in\mathcal{I}} x_f(u,i)\, E_f^{\mathrm{Off}}(u,i), \quad \forall u = \mathrm{st}_f. \tag{17}$$

IV. Problem Formulation and GBD Approach

A. Problem Formulation

Considering association strategy $\mathbf{x}$, data amount processed at satellites $\mathbf{y}$, users' transmit power $\mathbf{p}$, computing resources $\mathbf{n} = \left\{n_f^{\mathrm{LC}}(u), n_f^{\mathrm{EC}}(i) \middle| u\in\mathcal{U}, i\in\mathcal{I}, f\in\mathcal{F}\right\}$ and offloading ratio $\boldsymbol{\rho}$, we formulate a multi-tier partial computation offloading problem to jointly minimize the weighted sum of E2E delay and energy consumption as follows.

$$\mathcal{P}0: \min_{\mathbf{x},\mathbf{y},\mathbf{p},\mathbf{n},\boldsymbol{\rho}} U(\mathbf{x},\mathbf{y},\mathbf{p},\mathbf{n},\boldsymbol{\rho})$$

$$= \sum_{f\in\mathcal{F}}\left[\eta\delta_T T_f^{\mathrm{total}}(u) + (1-\eta)\,\delta_E E_f^{\mathrm{total}}(u)\right] \tag{18a}$$

$$\text{s.t. C1: } x_f(u,i)\in\{0,1\}, \quad \forall i\in\mathcal{I}, f\in\mathcal{F}, u=\mathrm{st}_f, \tag{18b}$$

$$\text{C2: } \sum_{i\in\mathcal{I}} x_f(u,i) \le 1, \quad \forall f\in\mathcal{F}, u=\mathrm{st}_f, \tag{18c}$$

$$\text{C3: } 0 \le x_f(u,i)\, p_f^{\mathrm{TX}}(u,i) \le P_{\max}(u,i), \quad \forall i\in\mathcal{I}, f\in\mathcal{F}, u=\mathrm{st}_f, \tag{18d}$$

$$\text{C4: } R_f(u,i) \ge x_f(u,i)\, R_{\mathrm{th},f}, \quad \forall i\in\mathcal{I},\ f\in\mathcal{F}, u=\mathrm{st}_f, \tag{18e}$$

$$\text{C5: } x_f(u,i)(1-\rho_f) b_f R_p \le R_f(u,i), \quad \forall i\in\mathcal{I}, f\in\mathcal{F}, u=\mathrm{st}_f, \tag{18f}$$

$$\text{C6: } 0 \le n_f^{\text{LC}}(u) \le N_{\max}(u), \quad \forall f \in \mathcal{F}, u = \text{st}_f, \tag{18g}$$

$$\text{C7: } \sum_{f \in \mathcal{F}} n_f^{\text{EC}}(i) \le N_{\max}(i), \quad \forall i \in \mathcal{I}, \tag{18h}$$

$$\text{C8: } n_f^{\text{EC}}(i) \ge 0, \quad \forall i \in \mathcal{I}, \ f \in \mathcal{F}, \tag{18i}$$

$$\text{C9: } 0 \le y_f^{\text{EC}}(s) \le x_f(u,s)(1-\rho_f) b_f, \ \forall f \in \mathcal{F}, \ u = \text{st}_f, \tag{18j}$$

$$\text{C10: } \rho_f \in [0,1], \quad \forall f \in \mathcal{F}, \tag{18k}$$

$$\text{C11: } T_f^{\text{total}}(u) \le T_f^{\max}, \quad \forall f \in \mathcal{F}, \ u = \text{st}_f. \tag{18l}$$

Here, $\eta \in [0,1]$ is the balancing factor defining the relative weight of delay and energy. δ_T and δ_E are normalized factors making the same range for these two metrics [41]. C1 and C2 mean that user association variable is binary, and each user can offload its task to at most one SAP. C3 ensures that non-negative user's transmit power cannot exceed its maximum value, and there is no transmit power when the user is not associated with the SAP. C4 guarantees the achievable rate of the user-to-SAP link is not less than the threshold value, and C5 means the offloaded data should not exceed the capacity. C6–C8 represent that the results of the allocated computation resources are non-negative and cannot exceed their computation capacity. C9 constraints the range of the packets offloaded to SN for MEC. C10 denotes a partial offloading strategy. C11 represents that the E2E delay of each task should not exceed its maximum delay tolerance.

B. Feasibility Analysis

First, we discuss the upper bound of the locally processed ratio ρ_f. From (18k), (18g) and (18l), we have $\rho_f \le \min\left\{1, \frac{z_{f,0}}{b_f}\right\}$, where $z_{f,0} = \frac{T_f^{\max} N_{\max}(u)}{\omega_u R_p}$. When $b_f \le z_{f,0}$, $(\rho_f)_{\max} = 1$ holds, indicating that task f can be processed at the user completely. When $b_f > z_{f,0}$, $(\rho_f)_{\max} = \frac{z_{f,0}}{b_f}$ holds, denoting that the user must offload its data to another SAP.

Then, we discuss the lower bound of ρ_f.

Proposition 1: The lower bound of ρ_f satisfies

$$\rho_f \ge \max\left\{0, 1 - \frac{z_{f,1}}{b_f}, 1 - \frac{z_{f,2}}{b_f}\right\}, \tag{19}$$

where $z_{f,1} = \frac{R_f(u,i)}{x_f(u,i) R_p}$, *if* $x_f(u,i) = 1$ *and*

$$z_{f,2} = \begin{cases} \dfrac{\frac{T_f^{\max}}{x_f(u,a)} - T_f^{\text{AN,pro}}(u,a)}{R_p\left(\frac{1}{R_f(u,a)} + \frac{\omega_a}{N_{\max}(a)} + \frac{\alpha}{R_f(a,u)}\right)}, \\ \qquad \text{if } x_f(u,a) = 1, \\ \min\left\{ \dfrac{\frac{T_f^{\max}}{x_f(u,s)} - T_f^{\text{SN,pro},1}(u,s)}{R_p\left[\frac{1}{R_f(u,s)} + \frac{\omega_s}{N_{\max}(s)} + \frac{\alpha}{R_f(s,u)}\right]}, \dfrac{\frac{T_f^{\max}}{x_f(u,s)} - T_f^{\text{SN,pro},2}(u,s)}{R_p\left(\frac{1}{R_f(u,s)} + C_{f,1}(u,s)\right)} \right\}, \\ \qquad \text{if } x_f(u,s) = 1. \end{cases} \tag{20}$$

Proof: Please see Appendix A. ∎

With different data amount b_f, $\mathcal{P}0$ is feasible if and only if ρ_f satisfies the following expressions.

$$\rho_f \in \begin{cases} [0,1], \text{if } b_f \le \min\{z_{f,0}, z_{f,1}, z_{f,2}\}, \\ \left[\max\left\{1 - \frac{z_{f,1}}{b_f}, 1 - \frac{z_{f,2}}{b_f}\right\}, 1\right], \\ \qquad \text{if } \min\{z_{f,1}, z_{f,2}\} \le b_f \le z_{f,0}, \\ \left[0, \frac{z_{f,0}}{b_f}\right], \text{if } z_{f,0} < b_f \le \min\{z_{f,1}, z_{f,2}\}, \\ \left[\max\left\{1 - \frac{z_{f,1}}{b_f}, 1 - \frac{z_{f,2}}{b_f}\right\}, \frac{z_{f,0}}{b_f}\right], \\ \text{if } \max\{z_{f,0}, z_{f,1}, z_{f,2}\} < b_f \le z_{f,0} + \min\{z_{f,1}, z_{f,2}\}, \\ \emptyset, \text{otherwise}. \end{cases} \tag{21}$$

C. Problem Decomposition Under GBD Approach

The structure of the MAX function makes the optimization problem tough to solve. Thus, we introduce an auxiliary variable to transform $\mathcal{P}0$. Define

$$r_f(u,s) \triangleq \max\{r_{f,1}(u,s), r_{f,2}(u,s)\}, \tag{22}$$

where

$$\begin{aligned} r_{f,1}(u,s) &= y_f^{\text{EC}}(s) R_p \left(\frac{\omega_s}{n_f^{\text{EC}}(s)} + \frac{\alpha}{R_f(s,u)}\right) \\ &\quad + T_f^{\text{SN,pro},1}(u,s), \\ r_{f,2}(u,s) &= \left[(1-\rho_f) b_f - y_f^{\text{EC}}(s)\right] R_p C_{f,1}(u,s) \\ &\quad + T_f^{\text{SN,pro},2}(u,s), \end{aligned}$$

and

$$C_{f,1}(u,s) = (1+\alpha) n_f^{\text{hop}} + \frac{1}{R_f(s_{\max}, g)} + \frac{\alpha}{R_f(g, s_{\max})} + \frac{\alpha}{R_f(s,u)}.$$

Define $\hat{T}_f^{\text{Off}}(u) \triangleq \sum_{i \in \mathcal{I}} x_f(u,i) \hat{T}_f^{\text{Off}}(u,i)$, where $\hat{T}_f^{\text{Off}}(u,s) = \frac{(1-\rho_f) b_f R_p}{R_f(u,s)} + r_f(u,s)$ and $\hat{T}_f^{\text{Off}}(u,a) = T_f^{\text{Off}}(u,a)$. Also, auxiliary variables $r_f(u)$ is used for each task f, which is defined as $r_f(u) \triangleq \max\left\{T_f^{\text{LC}}(u), \hat{T}_f^{\text{Off}}(u)\right\}$. Let $\mathbf{r} = \{r_f(u), r_f(u,s) | \, u \in \mathcal{U}, s \in \mathcal{S}, f \in \mathcal{F}\}$ denote the matrix containing the auxiliary variables. By linearizing the total delay term in (18a) and (18l), $\mathcal{P}_0$ can be transformed into

$$\begin{aligned} \mathcal{P}1: \ & \min_{\mathbf{x},\mathbf{y},\mathbf{p},\mathbf{n},\boldsymbol{\rho},\mathbf{r}} U(\mathbf{x},\mathbf{y},\mathbf{p},\mathbf{n},\boldsymbol{\rho},\mathbf{r}) \\ & = \sum_{f \in \mathcal{F}} \left[\eta \delta_T r_f(u) + (1-\eta) \delta_E E_f^{\text{total}}(u)\right] && \text{(23a)} \\ & \text{s.t. } r_f(u) \ge T_f^{\text{LC}}(u), \ \forall f \in \mathcal{F}, u = \text{st}_f, && \text{(23b)} \\ & \quad r_f(u) \ge \hat{T}_f^{\text{Off}}(u), \ \forall f \in \mathcal{F}, u = \text{st}_f, && \text{(23c)} \\ & \quad r_f(u) \le T_f^{\max}, \ \forall f \in \mathcal{F}, u = \text{st}_f, && \text{(23d)} \\ & \quad r_f(u,s) \ge r_{f,1}(u,s), \ r_f(u,s) \ge r_{f,2}(u,s), \\ & \quad \forall s \in \mathcal{S}, f \in \mathcal{F}, u = \text{st}_f, && \text{(23e)} \\ & \quad \text{(18b)} - \text{(18j)}, \text{(21)}. && \text{(23f)} \end{aligned}$$

$\mathcal{P}1$ is a mixed-integer nonlinear programming (MINLP) problem, which is an NP-hard problem due to the coupling relationship among different variables. GBD method is a common approach to tackle this kind of problem [42]. The main idea of GBD is to decompose the original problem into a primal problem and a master problem and then solve them iteratively. At each iteration, the primal problem is solved first and generates a result added to the master problem. Then, the master problem is solved. The upper bound obtained by the primal problem and the lower bound obtained by the master problem finally approach the optimal solutions after finite iterations.

Define $\boldsymbol{\phi} \triangleq \{\mathbf{x}, \mathbf{y}, \mathbf{n}, \boldsymbol{\rho}, \mathbf{r}\}$. $\mathcal{P}1$ is decomposed into two subproblems, i.e., $U(\mathbf{x}, \mathbf{y}, \mathbf{p}, \mathbf{n}, \boldsymbol{\rho}, \mathbf{r}) = U_1(\mathbf{p}) + U_2(\boldsymbol{\phi})$, where $U_1(\mathbf{p}) = (1-\eta)\,\delta_E \sum_{f\in\mathcal{F}} \sum_{i\in\mathcal{I}} x_f(u,i)\, E_f(u,i)$ is the primal problem which answers the question about how much power is required to offload tasks from users to SAPs. The remaining part

$$U_2(\boldsymbol{\phi}) = \sum_{f\in\mathcal{F}} \left[\eta\delta_T r_f(u) + (1-\eta)\,\delta_E \left(E_f^{\mathrm{LC}}(u) + \sum_{i\in\mathcal{I}} U_f(u,i) \right) \right]$$

is the master problem which reflects the question about association strategy, offloading proportion and resource allocation schemes, where $U_f(u,i) \triangleq x_f(u,i)\left(E_f^{\mathrm{Off}}(u,i) - E_f(u,i)\right)$.

V. Problem Transformation and MPTO Algorithm

A. Primal Problem

When solving the primal problem at l-th iteration, $\boldsymbol{\phi}^{(l)} = \left\{\mathbf{x}^{(l)}, \mathbf{y}^{(l)}, \mathbf{n}^{(l)}, \boldsymbol{\rho}^{(l)}, \mathbf{r}^{(l)}\right\}$ is fixed, and the only variable is $p_f^{\mathrm{TX}}(u,i)$. Since $R_f(u,i)$ is a monotonically increasing function of $p_f^{\mathrm{TX}}(u,i)$, $p_f^{\mathrm{TX}}(u,i)$ can be replaced by $R_f(u,i)$ with $p_f^{\mathrm{TX}}(u,i) = \frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)}$, where $C_{f,2}(u,i) = \frac{G_T(u)G_R(i)\left(\frac{\lambda(u,i)}{4\pi d_f(u,i)}\right)^2 L_a(u,i)|h_f(u,i)|^2}{k_B T_{\mathrm{no}}(i) W_{\mathrm{UL}}(u,i)}$. Therefore, the energy consumption when uploading tasks from user u to SAP i can be recast as $E_f(u,i) = \frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)R_f(u,i)} + \frac{p^{\mathrm{RX}}(u,i)}{R_f(u,i)}$. Define a function $f(x) \triangleq \frac{2^{\frac{x}{a}}-1}{cx}$. It can be proved that $\frac{\partial^2 f(x)}{\partial x^2} > 0$ when $x > 0$. Thus, $E_f(u,i)$ is the sum of two convex functions of $R_f(u,i)$. Then, the primal problem is equivalent recast as

$$\mathcal{P}2: \min_{\mathbf{R}} U_1(\mathbf{R}) = (1-\eta)\,\delta_E \sum_{f\in\mathcal{F}} (1-\rho_f)\, b_f R_p \cdot \sum_{i\in\mathcal{I}} x_f(u,i) \left(\frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)\,R_f(u,i)} + \frac{p^{\mathrm{RX}}(u,i)}{R_f(u,i)} \right) \tag{24a}$$

$$\text{s.t. } 0 \le x_f(u,i)\,\frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)} \le P_{\max}(u,i), \tag{24b}$$

$$(18e), (18f), (23c)\ . \tag{24c}$$

Since $\boldsymbol{\phi}^{(l)}$ is related to the solutions of $\mathcal{P}2$, the impact of $\boldsymbol{\phi}^{(l)}$ on the feasibility of $\mathcal{P}2$ should be discussed.

1) Feasible Case: Suppose that $\mathcal{P}2$ is feasible at l_1-th iteration. Since $\mathcal{P}2$ is a convex problem of $\mathbf{R}$, its partial Lagrangian function can be given by

$$\begin{aligned}
&\mathcal{L}\left(\mathbf{R}, \boldsymbol{\lambda}, \boldsymbol{\phi}^{(l_1)}\right) \\
&= U_1(\mathbf{R}) \\
&\quad + \sum_{f\in\mathcal{F}} \lambda_f^1 \left(\sum_{i\in\mathcal{I}} x_f^{(l_1)}(u,i) \hat{T}_f^{\mathrm{Off}}(u,i) - r_f^{(l_1)}(u) \right) \\
&\quad + \sum_{f\in\mathcal{F}} \sum_{i\in\mathcal{I}} \lambda_f^2(i) \left(x_f^{(l_1)}(u,i) \frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)} - P_{\max}(u,i) \right) \\
&\quad + \sum_{f\in\mathcal{F}} \sum_{i\in\mathcal{I}} \lambda_f^3(i) \left(x_f^{(l_1)}(u,i) R_{\mathrm{th},f} - R_f(u,i) \right) \\
&\quad + \sum_{f\in\mathcal{F}} \sum_{i\in\mathcal{I}} \lambda_f^4(i) \left[x_f^{(l_1)}(u,i) \left(1-\rho_f^{(l_1)}\right) b_f R_p - R_f(u,i) \right],
\end{aligned} \tag{25}$$

where $\boldsymbol{\lambda} = \left\{\lambda_f^1, \lambda_f^2(i), \lambda_f^3(i), \lambda_f^4(i)\right\} \ge 0$ is the Lagrange multiplier vector associated with $\mathcal{P}2$. Then, the Lagrangian dual problem of $\mathcal{P}2$ can be obtained by $\max_{\boldsymbol{\lambda}} \min_{\mathbf{R}} \mathcal{L}\left(\mathbf{R}, \boldsymbol{\lambda}, \boldsymbol{\phi}^{(l_1)}\right)$. The optimal solution $\mathbf{R}^{(l_1)}$ (i.e., $\mathbf{p}^{(l_1)}$) and $\boldsymbol{\lambda}^{(l_1)}$ can be obtained by applying Karush–Kuhn–Tucker (KKT) conditions since strong duality holds.

2) Infeasible Case: Assume that at l_2-th iteration $\mathcal{P}2$ is infeasible, which is caused by the violations of the constraints (24b)-(24c) related to $\boldsymbol{\phi}^{(l_2)}$. In this case, the feasibility-check problem of the primal problem must exist feasible points [42]. An auxiliary variable Q is introduced to construct a feasibility-check problem $P2'$ given by

$$\mathcal{P}2': \min_{Q,\mathbf{R}} Q \tag{26a}$$

$$\text{s.t. } \sum_{i\in\mathcal{I}} x_f(u,i)\,\hat{T}_f^{\mathrm{Off}}(u,i) - r_f(u) \le Q,\ \forall f \in \mathcal{F}, \tag{26b}$$

$$x_f(u,i)\,\frac{2^{\frac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1}{C_{f,2}(u,i)} - P_{\max}(u,i) \le Q, \quad \forall i \in \mathcal{I},\ f \in \mathcal{F},\ u = \mathrm{st}_f, \tag{26c}$$

$$x_f(u,i)\,R_{\mathrm{th},f} - R_f(u,i) \le Q, \quad \forall i \in \mathcal{I},\ f \in \mathcal{F},\ u = \mathrm{st}_f, \tag{26d}$$

$$x_f(u,i)\,(1-\rho_f)\,b_f R_p - R_f(u,i) \le Q, \quad \forall i \in \mathcal{I},\ f \in \mathcal{F},\ u = \mathrm{st}_f. \tag{26e}$$

Note that $P2'$ is equivalent to the feasible case of $\mathcal{P}2$ when $Q = 0$. Thus, we only discuss the infeasible case of $\mathcal{P}2$ here, that is, $Q \ne 0$. Since both the objective function and constraints related Q are convex, $\mathcal{P}2'$ is also a convex problem. Let $\hat{\mathcal{L}}\left(\mathbf{R}, \boldsymbol{\mu}, \boldsymbol{\phi}^{(l_2)}\right)$ denote the partial Lagrangian function of $\mathcal{P}2'$, where $\boldsymbol{\mu} = \left\{\mu_f^1, \mu_f^2(i), \mu_f^3(i), \mu_f^4(i)\right\} \ge 0$ is the Lagrange multiplier matrix associated with $\mathcal{P}2'$. Let $Q^{(l_2)}, \mathbf{R}^{(l_2)}, \boldsymbol{\mu}^{(l_2)}$ denote the optimal solution of $\mathcal{P}2'$ and its dual variables at l_2-th iteration, which can be obtained by $\left\{Q^{(l_2)}, \mathbf{R}^{(l_2)}\right\} = \arg\min_{Q,\mathbf{R}} \hat{\mathcal{L}}\left(\mathbf{R}, \boldsymbol{\mu}^{(l_2)}, \boldsymbol{\phi}^{(l_2)}\right)$.

With KKT condition $\frac{\partial \hat{\mathcal{L}}\left(\mathbf{R},\boldsymbol{\mu}^{(l_2)},\boldsymbol{\phi}^{(l_2)}\right)}{\partial Q}=0$, the optimal Lagrangian multipliers should satisfy $\sum_{f\in\mathcal{F}}\mu_f^{1(l_2)}+\sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}}\left(\mu_f^{2(l_2)}(i)+\mu_f^{3(l_2)}(i)+\mu_f^{4(l_2)}(i)\right)=1$, and the optimal capacity of user-to-SAP links satisfies

$$\mathbf{R}^{(l_2)} = \arg\min_{\mathbf{R}} \sum_{f\in\mathcal{F}}\sum_{i\in\mathcal{I}} \begin{bmatrix} \mu_f^{1(l_2)} x_f^{(l_2)}(u,i)\dfrac{\left(1-\rho_f^{(l_2)}\right) b_f R_p}{R_f(u,i)} \\ +\dfrac{\mu_f^{2(l_2)}(i)\, x_f^{(l_2)}(u,i)}{C_{f,2}^{(l_2)}(u,i)} \\ \left(2^{\dfrac{R_f(u,i)}{W_{\mathrm{UL}}(u,i)}}-1\right) \\ -\left(\mu_f^{3(l_2)}(i)+\mu_f^{4(l_2)}(i)\right) R_f(u,i) \end{bmatrix}. \tag{27}$$

Then, the optimal power allocation $\mathbf{p}^{(l_2)}$ can also be obtained.

B. Master Problem

The variable $\mathbf{R}$ is fixed when discussing the master problem. Since all the variables in $\boldsymbol{\phi}$ are continuous except $\mathbf{x}$, the relaxation-based method is often used to relax the binary variables into continuous ones. However, it is not suitable to relax directly because $\mathbf{x}$ will be the same as $\boldsymbol{\rho}$ in this case and lose its physical meaning. Since $x_f^2(u,i)\le x_f(u,i)$ when $x_f(u,i)\in[0,1]$ and the equal sign holds iff $x_f(u,i)\in\{0,1\}$, the constraint $x_f^2(u,i)=x_f(u,i)$ should be added when replacing the binary variables $x_f(u,i)$ into continuous form $x_f(u,i)\in[0,1]$.

Suppose that the upper bound of $\mathcal{L}\left(\mathbf{R}^{(l_1)},\boldsymbol{\lambda}^{(l_1)},\boldsymbol{\phi}\right)$ is M, which is an auxiliary variable to formulate the master problem [42]. Then, the master problem can be given by

$$\mathcal{P}3:\ \min_{M,\boldsymbol{\phi}}\ M+U_2(\boldsymbol{\phi}) \tag{28a}$$

$$\text{s.t. } \mathcal{L}\left(\mathbf{R}^{(l_1)},\boldsymbol{\lambda}^{(l_1)},\boldsymbol{\phi}\right)\le M,\ \forall l_1\in\{1,\dots,L_1\}, \tag{28b}$$

$$\hat{\mathcal{L}}\left(\mathbf{R}^{(l_2)},\boldsymbol{\mu}^{(l_2)},\boldsymbol{\phi}\right)\le 0,\ \forall l_2\in\{1,\dots,L_2\}, \tag{28c}$$

$$x_f(u,i)\in[0,1],\ \forall i\in\mathcal{I}, f\in\mathcal{F}, u=\mathrm{st}_f, \tag{28d}$$

$$x_f^2(u,i)-x_f(u,i)=0,\ \forall i\in\mathcal{I}, f\in\mathcal{F}, u=\mathrm{st}_f, \tag{28e}$$

$$\text{(23b), (23d), (23e), (23f),} \tag{28f}$$

where L_1 and L_2 denote the total number of iterations in which $\mathcal{P}2$ is feasible or not. At l-th iteration of GBD method, $L_1+L_2=l$ holds. The master problem $\mathcal{P}3$ is the relaxed form of the original problem $\mathcal{P}1$ [42]. Thus, whether $\mathcal{P}3$ is feasible is equivalent to that of $\mathcal{P}1$. Since we have discussed the feasibility of $\mathcal{P}1$ in Section IV-B, $\mathcal{P}1$ is regarded as feasible in this paper.

C. Modified Parallel SCA Algorithm

$\mathcal{P}3$ is an NP-hard problem due to multiple coupling parameters. The parallel SCA algorithm can solve the multi-variable non-convex problem, which does not require the convexity of the objective function and constraints [43]. With multiple approximation choices, the parallel SCA algorithm offers much flexibility to transform the non-convex optimization problem into approximate convex terms and can achieve convergence after limited iterations [44]. Based on the SCA theory, we extend the existing SCA approximation methods and transform the master problem into a tractable convex problem.

Theorem 1: Based on the Example 8 in [44], we can deduce that when $g(\mathbf{z})$ has a product of functions (PF) structure given by $g(\mathbf{z})=g_1(\mathbf{z})\,g_2(\mathbf{z})\,g_3(\mathbf{z})$, where $g_1(\mathbf{z})$ and $g_2(\mathbf{z})$ are both positive and convex functions in the feasible set $\mathcal{Z}$. The gradient of g can be obtained by $\nabla_{\mathbf{z}}g=g_2g_3\nabla_{\mathbf{z}}g_1+g_1g_3\nabla_{\mathbf{z}}g_2+g_1g_2\nabla_{\mathbf{z}}g_3$. Thus, for any $\mathbf{z}^{(k)}\in\mathcal{Z}$, a convex approximation of $g(\mathbf{z})$ can be expressed as

$$\begin{aligned}&\dot{g}\left(\mathbf{z};\mathbf{z}^{(k)}\right)\\ &=g_1(\mathbf{z})\,g_2\left(\mathbf{z}^{(k)}\right)g_3\left(\mathbf{z}^{(k)}\right)\\ &\quad+g_1\left(\mathbf{z}^{(k)}\right)g_2(\mathbf{z})\,g_3\left(\mathbf{z}^{(k)}\right)+g_1\left(\mathbf{z}^{(k)}\right)g_2\left(\mathbf{z}^{(k)}\right)g_3(\mathbf{z})\\ &\quad+\frac{\boldsymbol{\Lambda}}{2}\left(\mathbf{z}-\mathbf{z}^{(k)}\right)^{\mathrm{T}}\mathbf{H}\left(\mathbf{z}^{(k)}\right)\left(\mathbf{z}-\mathbf{z}^{(k)}\right),\end{aligned} \tag{29}$$

where $\boldsymbol{\Lambda}$ is a positive constant matrix. $\mathbf{H}(\cdot)$ is a uniformly positive definite matrix, which could be omitted if g_1, g_2 and g_3 are bounded away from zero on $\mathcal{Z}$ and $g_1+g_2+g_3$ is strongly convex.

Theorem 2: Suppose that $g(\mathbf{z})$ has a PF structure given by $g(\mathbf{z})=g_1(\mathbf{z})\,g_2(\mathbf{z})$ ($\mathbf{z}\in\mathcal{Z}$), where g_1 and g_2 are positive but not convex. The following approximation $\dot{g}\left(\mathbf{z};\mathbf{z}^{(k)}\right)=\ddot{g}_1\left(\mathbf{z};\mathbf{z}^{(k)}\right)g_2\left(\mathbf{z}^{(k)}\right)+g_1\left(\mathbf{z}^{(k)}\right)\ddot{g}_2\left(\mathbf{z};\mathbf{z}^{(k)}\right)$ can be utilized, where $\ddot{g}_1$ and $\ddot{g}_2$ are any approximation for g_1 and g_2. Suppose that $\ddot{g}_1$ is the sum of multiple subfunctions (note that any subfunction cannot be necessarily convex) with $\ddot{g}_1\left(\mathbf{z};\mathbf{z}^{(k)}\right)=\sum_i\ddot{g}_1^i\left(\mathbf{z}_i;\mathbf{z}^{(k)}\right)$, the gradient-like approximation can be utilized, with $\ddot{g}_1^i\left(\mathbf{z}_i;\mathbf{z}^{(k)}\right)=\nabla_{\mathbf{z}_i}g_1\left(\mathbf{z}^{(k)}\right)^{\mathrm{T}}\left(\mathbf{z}_i-\mathbf{z}_i^{(k)}\right)+\frac{\Lambda_i}{2}\left\|\mathbf{z}_i-\mathbf{z}_i^{(k)}\right\|^2$ (Example 6, [44]).

Note that $U_f(u,a)=x_f(u,a)\left(E_f^{\mathrm{EC}}(a)+E_f(a,u)\right)$ holds. According to Theorem 1, for the given feasible solution $\boldsymbol{\phi}^{(k)}=\left\{\mathbf{x}^{(k)},\mathbf{y}^{(k)},\mathbf{n}^{(k)},\boldsymbol{\rho}^{(k)},\mathbf{r}^{(k)}\right\}$ at the k-th iteration of the SCA algorithm, the convex approximation of $U_f(u,a)$ can be obtained by

$$\begin{aligned}&\dot{U}_f\left(u,a|\,x_f(u,a),\rho_f,n_f^{\mathrm{EC}}(a);x_f^{(k)}(u,a),\rho_f^{(k)},n_f^{\mathrm{EC}(k)}(a)\right)\\ &\triangleq\kappa_a\omega_a b_f R_p\begin{bmatrix} x_f(u,a)\left(1-\rho_f^{(k)}\right)\left(n_f^{\mathrm{EC}(k)}(a)\right)^2\\ +x_f^{(k)}(u,a)(1-\rho_f)\left(n_f^{\mathrm{EC}(k)}(a)\right)^2\\ +x_f^{(k)}(u,a)\left(1-\rho_f^{(k)}\right)\left(n_f^{\mathrm{EC}}(a)\right)^2\\ +\dfrac{\Lambda_{\rho_f}}{2}\left(\rho_f-\rho_f^{(k)}\right)^2\\ +\dfrac{\Lambda_{x_f(a)}}{2}\left(x_f(u,a)-x_f^{(k)}(u,a)\right)^2\\ +\dfrac{\Lambda_{n_f^{\mathrm{EC}}(a)}}{2}\left(n_f^{\mathrm{EC}}(a)-n_f^{\mathrm{EC}(k)}(a)\right)^2\end{bmatrix}\end{aligned}$$

$$+\frac{\alpha R_p\left(p^{\mathrm{TX}}(a,u)+p^{\mathrm{RX}}(a,u)\right)}{R_f(a,u)} \times \dot{y}_f\left(u,a|\, x_f(u,a),\rho_f; x_f^{(k)}(u,a),\rho_f^{(k)}\right), \tag{30}$$

where

$$\dot{y}_f\left(u,a|\, x_f(u,a),\rho_f; x_f^{(k)}(u,a),\rho_f^{(k)}\right) \triangleq b_f\begin{bmatrix} x_f(u,a)\left(1-\rho_f^{(k)}\right)+x_f^{(k)}(u,a)(1-\rho_f) \\ +\frac{\Lambda_{\rho_f}}{2}\left(\rho_f-\rho_f^{(k)}\right)^2 \\ +\frac{\Lambda_{x_f(a)}}{2}\left(x_f(u,a)-x_f^{(k)}(u,a)\right)^2 \end{bmatrix}. \tag{31}$$

Let $\dot{\mathcal{L}}\left(\mathbf{R}^{(l_1)},\boldsymbol{\lambda}^{(l_1)},\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right)$, $\dot{\hat{\mathcal{L}}}\left(\mathbf{R}^{(l_2)},\boldsymbol{\mu}^{(l_2)},\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right)$, $\dot{T}_f^{\mathrm{LC}}\left(u|\,\rho_f,n_f^{\mathrm{LC}};\rho_f^{(k)},n_f^{\mathrm{LC}(k)}\right)$, $\dot{E}_f^{\mathrm{LC}}\left(u|\,\rho_f,n_f^{\mathrm{LC}};\rho_f^{(k)},n_f^{\mathrm{LC}(k)}\right)$ and $\dot{r}_{f,1}\left(u,s|\, y_f^{\mathrm{EC}}(s),n_f^{\mathrm{EC}}(s);y_f^{\mathrm{EC}(k)}(s),n_f^{\mathrm{EC}(k)}(s)\right)$ denote the convex approximation form of $\mathcal{L}\left(\mathbf{R}^{(l_1)},\boldsymbol{\lambda}^{(l_1)},\boldsymbol{\phi}\right)$, $\hat{\mathcal{L}}\left(\mathbf{R}^{(l_2)},\boldsymbol{\mu}^{(l_2)},\boldsymbol{\phi}\right)$, $T_f^{\mathrm{LC}}(u)$, $E_f^{\mathrm{LC}}(u)$ and $r_{f,1}(u,s)$ at the k-th iteration respectively, which can be obtained by the similar method with (30).

According to Theorem 2, the convex approximation function of $y_f^{\mathrm{CC}}(g)$ in $r_{f,2}(u,s)$ can be given by

$$\begin{aligned}\dot{y}_f^{\mathrm{CC}}&\left(g|\,\rho_f,y_f^{\mathrm{EC}}(s);\rho_f^{(k)},y_f^{\mathrm{EC}(k)}(s)\right) \\ &\triangleq b_f-b_f\left(\rho_f-\rho_f^{(k)}\right)+\frac{\Lambda_{\rho_f}}{2}\left(\rho_f-\rho_f^{(k)}\right)^2 \\ &\quad-\left(y_f^{\mathrm{EC}}(s)-y_f^{\mathrm{EC}(k)}(s)\right) \\ &\quad+\frac{\Lambda_{y_f^{\mathrm{EC}}(s)}}{2}\left(y_f^{\mathrm{EC}}(s)-y_f^{\mathrm{EC}(k)}(s)\right)^2,\end{aligned} \tag{32}$$

which is the part of the convex approximation function of $\dot{r}_{f,2}\left(u,s|\,\rho_f,y_f^{\mathrm{EC}}(s);\rho_f^{(k)},y_f^{\mathrm{EC}(k)}(s)\right)$. Then, the convex approximation of $U_f(u,s)$ can be obtained by (33), as shown at the bottom of the page.

Therefore, the convex approximation of $U_2(\boldsymbol{\phi})$ at k-th iteration is $\dot{U}_2\left(\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right) = \sum_{f\in\mathcal{F}}\left[\eta\delta_T r_f(u)+(1-\eta)\,\delta_E\cdot\left(\dot{E}_f^{\mathrm{LC}}(u)+\sum_{i\in\mathcal{I}}\dot{U}_f(u,i)\right)\right]$, and the convex approximation problem of $\mathcal{P}3$ at k-th iteration can be given by

$$\mathcal{P}3':\ \min_{M,\boldsymbol{\phi}}\ M+\dot{U}_2\left(\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right) \tag{34a}$$

$$\text{s.t. } \dot{\mathcal{L}}\left(\mathbf{R}^{(l_1)},\boldsymbol{\lambda}^{(l_1)},\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right)\le M,\ \forall l_1\in\{1,\dots,L_1\}, \tag{34b}$$

$$\dot{\hat{\mathcal{L}}}\left(\mathbf{R}^{(l_2)},\boldsymbol{\mu}^{(l_2)},\boldsymbol{\phi};\boldsymbol{\phi}^{(k)}\right)\le 0,\ \forall l_2\in\{1,\dots,L_2\}, \tag{34c}$$

$$r_f(u)\ge\dot{T}_f^{\mathrm{LC}}\left(u|\,\rho_f,n_f^{\mathrm{LC}};\rho_f^{(k)},n_f^{\mathrm{LC}(k)}\right), \tag{34d}$$

$$r_f(u,s)\ge\dot{r}_{f,1}\left(u,s|\, y_f^{\mathrm{EC}}(s),n_f^{\mathrm{EC}}(s);y_f^{\mathrm{EC}(k)}(s),n_f^{\mathrm{EC}(k)}(s)\right), \tag{34e}$$

$$r_f(u,s)\ge\dot{r}_{f,2}\left(u,s|\,\rho_f,y_f^{\mathrm{EC}}(s);\rho_f^{(k)},y_f^{\mathrm{EC}(k)}(s)\right), \tag{34f}$$

$$0\le y_f^{\mathrm{EC}}(s)\le\dot{y}_f\left(u,i|\, x_f(u,i),\rho_f;x_f^{(k)}(u,i),\rho_f^{(k)}\right), \tag{34g}$$

$$\text{(23f), (23d), (28d)-(28e).} \tag{34h}$$

Since $\mathcal{P}3'$ is a convex problem, it owns the unique solution denoted by $\boldsymbol{\phi}^*\left(\boldsymbol{\phi}^{(k)}\right)$. The convergence condition is that for any task f, the following expression holds,

$$\delta_f^{(k)}=\frac{\left\|x_f^{(k)}(u,i)-x_f^{(k-1)}(u,i)\right\|}{\left\|x_f^{(k-1)}(u,i)\right\|}+\frac{\left\|y_f^{\mathrm{EC}(k)}(s)-y_f^{\mathrm{EC}(k-1)}(s)\right\|}{\left\|y_f^{\mathrm{EC}(k-1)}(s)\right\|}$$

$$\begin{aligned}\dot{U}_f&\left(u,s|\, x_f(u,s),\rho_f,n_f^{\mathrm{EC}}(s),y_f^{\mathrm{EC}}(s);x_f^{(k)}(u,s),\rho_f^{(k)},n_f^{\mathrm{EC}(k)}(a),y_f^{\mathrm{EC}(k)}(s)\right) \\ &\triangleq\frac{\alpha\left(p^{\mathrm{TX}}(s,u)+p^{\mathrm{RX}}(s,u)\right)R_p}{R_f(s,u)}\dot{y}_f\left(u,s|\, x_f(u,s),\rho_f;x_f^{(k)}(u,s),\rho_f^{(k)}\right) \\ &\quad+(1+\alpha)R_p\left[\frac{2p_{\mathrm{ISL}}n_f^{\mathrm{hop}}}{R_{\mathrm{ISL}}}+\frac{\left(p^{\mathrm{TX}}(s,g)+p^{\mathrm{RX}}(s,g)\right)}{R_f^{\mathrm{S2G}}}\right]\begin{Bmatrix} x_f(u,s)\left[\left(1-\rho_f^{(k)}\right)b_f-y_f^{\mathrm{EC}(k)}(s)\right] \\ +x_f^{(k)}(u,s)\,\dot{y}_f^{\mathrm{CC}}\left(g|\,\rho_f,y_f^{\mathrm{EC}}(s);\rho_f^{(k)},y_f^{\mathrm{EC}(k)}(s)\right)\end{Bmatrix} \\ &\quad+\kappa_s\omega_s R_p\begin{bmatrix} x_f(u,s)\,y_f^{\mathrm{EC}(k)}(s)\left(n_f^{\mathrm{EC}(k)}(s)\right)^2+x_f^{(k)}(u,s)\,y_f^{\mathrm{EC}}(s)\left(n_f^{\mathrm{EC}(k)}(s)\right)^2 \\ +x_f^{(k)}(u,s)\,y_f^{\mathrm{EC}(k)}(s)\left(n_f^{\mathrm{EC}}(s)\right)^2+\frac{\Lambda_{x_f(s)}}{2}\left(x_f(u,s)-x_f^{(k)}(u,s)\right)^2 \\ +\frac{\Lambda_{y_f^{\mathrm{EC}}(s)}}{2}\left(y_f^{\mathrm{EC}}(s)-y_f^{\mathrm{EC}(k)}(s)\right)^2+\frac{\Lambda_{n_f^{\mathrm{EC}}(s)}}{2}\left(n_f^{\mathrm{EC}}(s)-n_f^{\mathrm{EC}(k)}(s)\right)^2\end{bmatrix} \\ &\quad+A_0\left\{x_f(u,s)\exp\left\{A_1\left[\left(1-\rho_f^{(k)}\right)b_f-y_f^{\mathrm{EC}(k)}(s)\right]\right\}+x_f^{(k)}(u,s)\exp\left(A_1\dot{y}_f^{\mathrm{CC}}\left(g|\,\rho_f,y_f^{\mathrm{EC}}(s);\rho_f^{(k)},y_f^{\mathrm{EC}(k)}(s)\right)\right)\right\}.\end{aligned} \tag{33}$$

Algorithm 1 MPTO Algorithm

Input: Maximum outer iteration number L and maximum inner iteration number of master problem K.
Output: Optimal transmit power $\mathbf{p}$, joint association strategy, resource and task allocation scheme $\boldsymbol{\phi}$.

Initialization: Initialize $\boldsymbol{\phi}=\boldsymbol{\phi}^{(0)}$, $\nu^{(0)}=1$, $l=0$, $l_1=0$, $l_2=0$, $\text{UBD}^{(0)}=\infty$, $\text{LBD}^{(0)}=-\infty$.
while $\left|\left(UBD^{(l)}-LBD^{(l)}\right)/LBD^{(l)}\right|>\epsilon_{GBD}$ and $l<L$ **do**
  Solve the primal problem $\mathcal{P}2$:
  Fix $\boldsymbol{\phi}$ as $\boldsymbol{\phi}^{(l)}$;
  $l=l+1$;
  if $\mathcal{P}2$ *is feasible* **then**
    $l_1=l_1+1$;
    Solve $\max_{\boldsymbol{\lambda}}\min_{\mathbf{R}}\mathcal{L}\left(\mathbf{R},\boldsymbol{\lambda},\boldsymbol{\phi}^{(l-1)}\right)$ and obtain optimal solution $\left\{\mathbf{R}^{(l)},\boldsymbol{\lambda}^{(l)}\right\}$;
    Update $\text{UBD}^{(l)}$ with $\text{UBD}^{(l)}=\min\left\{\mathcal{H}^{(l-1)}+U\left(\boldsymbol{\phi}^{(l-1)}\right),\text{UBD}^{(l-1)}\right\}$;
    Generate an optimality cut $\mathcal{L}\left(\mathbf{R}^{(l)},\boldsymbol{\lambda}^{(l)},\boldsymbol{\phi}\right)\le M$;
  end
  else
    $l_2=l_2+1$;
    Solve $\max_{\boldsymbol{\mu}}\min_{\mathbf{R},Q}\hat{\mathcal{L}}\left(\mathbf{R},\boldsymbol{\mu},\boldsymbol{\phi}^{(l-1)}\right)$ and obtain optimal solution $\left\{\mathbf{R}^{(l)},\boldsymbol{\mu}^{(l)}\right\}$;
    Generate a feasibility cut $\hat{\mathcal{L}}\left(\mathbf{R}^{(l)},\boldsymbol{\mu}^{(l)},\boldsymbol{\phi}\right)\le 0$;
  end
  Add a new cut $\mathcal{L}\left(\mathbf{R}^{(l)},\boldsymbol{\lambda}^{(l)},\boldsymbol{\phi}\right)\le M$ or $\hat{\mathcal{L}}\left(\mathbf{R}^{(l)},\boldsymbol{\mu}^{(l)},\boldsymbol{\phi}\right)\le 0$ into the master problem $\mathcal{P}3$;
  Solve the master problem $\mathcal{P}3$ *based on parallel SCA algorithm:*
  Set $\dot{\boldsymbol{\phi}}^{(0)}=\boldsymbol{\phi}^{(l-1)}$ and $k=0$;
  repeat
    Solve the problem $\mathcal{P}3'$ and obtain $\boldsymbol{\phi}^{*}\left(\dot{\boldsymbol{\phi}}^{(k)}\right)$ and $\dot{M}$;
    Update $\dot{\boldsymbol{\phi}}^{(k+1)}$ with $\dot{\boldsymbol{\phi}}^{(k+1)}=\dot{\boldsymbol{\phi}}^{(k)}+\nu^{k}\left(\boldsymbol{\phi}^{*}\left(\dot{\boldsymbol{\phi}}^{(k)}\right)-\dot{\boldsymbol{\phi}}^{(k)}\right)$;
    Update $\nu^{(k+1)}=\frac{\nu^{k}+w_1^{k}}{1+w_2^{k}}$;
    $k=k+1$;
  until $\forall f\in\mathcal{F},\delta_f^{(k)}\le\epsilon_f$ or $k>K$;
  Update $\boldsymbol{\phi}^{(l)}=\dot{\boldsymbol{\phi}}^{(k)}$ and $M^{(l)}=\dot{M}$;
  Update $\text{LBD}^{(l)}$ with $\text{LBD}^{(l)}=M^{(l)}+\dot{U}\left(\boldsymbol{\phi}^{(l)}\right)$;
end

$$+\frac{\left\|\mathbf{n}_f^{(k)}-\mathbf{n}_f^{(k-1)}\right\|}{\left\|\mathbf{n}_f^{(k-1)}\right\|}+\frac{\left\|\rho_f^{(k)}-\rho_f^{(k-1)}\right\|}{\left\|\rho_f^{(k-1)}\right\|}+\frac{\left\|\mathbf{r}_f^{(k)}-\mathbf{r}_f^{(k-1)}\right\|}{\left\|\mathbf{r}_f^{(k-1)}\right\|}\le\varepsilon_f. \tag{35}$$

D. MPTO Algorithm

Define $\mathcal{H}^{(l)}\triangleq\max_{\boldsymbol{\lambda}}\min_{\mathbf{R}}\mathcal{L}\left(\mathbf{R},\boldsymbol{\lambda},\boldsymbol{\phi}^{(l)}\right)$ as the optimal value of the Lagrange dual problem of $\mathcal{P}2$ at $(l+1)$-th iteration. The process of MPTO algorithm is shown in Algorithm 1.

E. Convergence and Complexity Analysis

1) Convergence Analysis of the Master Problem:

Theorem 3: Suppose that the objective function $F\left(\mathbf{z};\mathbf{z}^{(k)}\right)$ *is strongly convex and differentiable on its domain* $\mathcal{Z}$. *With parallel SCA algorithm, the update rule of variables is* $\mathbf{z}^{(k+1)}=\mathbf{z}^{(k)}+w^{(k)}\left(\mathbf{z}^{*}\left(\mathbf{z}^{(k)}\right)-\mathbf{z}^{(k)}\right)$, *where* $\mathbf{z}^{*}\left(\mathbf{z}^{(k)}\right)$ *is the optimal solution of* $F\left(\mathbf{z};\mathbf{z}^{(k)}\right)$. *When the diminishing stepsize* $w^{(k)}$ *satisfies* $\sum_{k=1}^{\infty}w^{(k)}=+\infty$ *and* $\sum_{k=1}^{\infty}\left(w^{(k)}\right)^2<+\infty$, *then the parallel SCA algorithm can converge to a stationary point, i.e.,* $\lim_{k\to\infty}\left\|\mathbf{z}^{*}\left(\mathbf{z}^{(k)}\right)-\mathbf{z}^{(k)}\right\|=0$ *[45].*

Considering that the diminishing stepsize rules follows $\nu^{(k+1)}=\frac{\nu^{(k)}+w_1^{(k)}}{1+w_2^{(k)}}$. In order to guarantee the convergence of the parallel SCA algorithm, $w_1^{(k)}$ and $w_2^{(k)}$ should satisfy $0\le w_1^{(k)}\le w_2^{(k)}$, $w_1^{(k)}/w_2^{(k)}\to 0$ when $k\to\infty$, while $\sum_k w_1^{(k)}/w_2^{(k)}=\infty$. Here, we adopt that $w_1^{(k)}=w_1$ and $w_2^{k}=w_2\cdot k$, where $w_1\in(0,1),w_2\in(0,1)$, and $w_1\le w_2$ hold.

2) Convergence Analysis of the Original Problem:

Proposition 2: At each iteration, the optimal objective value of $\mathcal{P}2$ *and* $\mathcal{P}3$ *are the upper and lower bounds of* $\mathcal{P}1$*'s optimal objective value, respectively.*

Proof: Upper bound: We desire to prove that $\text{UBD}^{(l)}=\min\left\{\mathcal{H}^{(l-1)}+U_2\left(\boldsymbol{\phi}^{(l-1)}\right),\text{UBD}^{(l-1)}\right\}$ is the upper bound of $\mathcal{P}1$. When $\mathcal{P}2$ is infeasible, $\mathcal{H}^{(l-1)}$ is infinite according to dual theory, which is indeed the upper bound of $\mathcal{P}1$. Then, the case that $\mathcal{P}2$ is feasible will be analyzed with the proof by contradiction. Let U^{*} denote the optimal value of $\mathcal{P}1$'s objective function. With proof by contradiction, $\text{UBD}^{(l)}\le U^{*}$ holds. Note that $\text{UBD}^{(l)}$ is the result after l-th iterations, it can be recast as $\text{UBD}^{(l)}=\min_{l'<l}\left\{\mathcal{H}^{(l')}+U_2\left(\boldsymbol{\phi}^{(l')}\right)\right\}$, where $\hat{l}<l$ denotes the iteration number which minimizes UBD. Due to the convexity of $\mathcal{P}2$, the strong duality holds, i.e., $\mathcal{H}^{(l)}=U_1\left(\mathbf{R}^{(l)}\right)$. Then, it can be deduced that

$$\begin{aligned}\text{UBD}^{(l)}&=\mathcal{H}^{(\hat{l})}+U_2\left(\boldsymbol{\phi}^{(\hat{l})}\right)=U_1\left(\mathbf{R}^{(\hat{l})}\right)+U_2\left(\boldsymbol{\phi}^{(\hat{l})}\right)\\&\le U_1\left(\mathbf{R}^{(*)}\right)+U_2\left(\boldsymbol{\phi}^{(*)}\right)=U^{*},\end{aligned}\tag{36}$$

where $*$ is the iteration number corresponding to U^{*}. However, (36) violates the assumption that U^{*} is the optimal value of $\mathcal{P}1$. Therefore, $\text{UBD}^{(l)}>U^{*}$ holds and $\text{UBD}^{(l)}$ is the upper bound of $\mathcal{P}1$.

Lower Bound: Let $\text{LBD}^{(l)}=M^{(l)}+U_2\left(\boldsymbol{\phi}^{(l)}\right)$ denote the optimal value of $\mathcal{P}3$ at l-th iteration. Then, we desire to prove that $\text{LBD}^{(l)}$ is the lower bound of $\mathcal{P}1$ at l-th iteration.

Since the primal problem $\mathcal{P}2$ is a convex problem, the original problem $\mathcal{P}1$ is equivalent to $\min_{\boldsymbol{\phi}}\left(\max_{\boldsymbol{\lambda}}\min_{\mathbf{R}}\mathcal{L}\left(\mathbf{R},\boldsymbol{\lambda},\boldsymbol{\phi}\right)\right)+U_2\left(\boldsymbol{\phi}\right)$ with constraints (23b)-(23f). Recall that $\mathcal{P}3$ is the relaxed form of $\mathcal{P}1$ with larger solution space due to the optimality and feasibility cuts. The objective function of $\mathcal{P}3$ at l-th iteration can be recast as $\min_{\boldsymbol{\phi}}\left(\sup\ \mathcal{L}\left(\mathbf{R},\boldsymbol{\lambda},\boldsymbol{\phi}^{(l-1)}\right)\right)+U_2\left(\boldsymbol{\phi}\right)$. Thus, the optimal value of $\mathcal{P}3$ is smaller than that of $\mathcal{P}1$. That is, $\text{LBD}^{(l)}$ is the lower bound of U^{*}. ■

TABLE III
MAIN SIMULATION PARAMETERS

Parameter	Value
$L_a(u,a), L_a(u,s)$	2.5 dB, 5.2 dB [19]
$G_T(u), G_T(a), G_T(s)$	0 dB, 40 dB, 53 dB
$K_f(u,a), K_f(u,s)$	[15,20] dB, [15.64,20] dB
$c/\lambda(u,a), c/\lambda(u,s)$	60 GHz, 30 GHz
n_0	-174 dBm/Hz
δ_T, δ_E	0.2, 2 $\times 10^{-3}$ [46]
b_f	[0.5,3] Mbits [28]
R_p	1080 [47]
κ_u, κ_i	1×10^{-26} [48], 1×10^{-28} [27]
$\omega_u, \omega_a, \omega_s$	40 [49], 100, 1000 [29]
$N_{\max}(u), N_{\max}(a), N_{\max}(s)$	4×10^8, 6×10^9, 1×10^{10} cycles/s
$T_f^{\max}$	0.5 s
$R_{\text{ISL}}, R_f^{\text{S2G}}$	10 Mbps, 20 Mbps [50]

3) Complexity Analysis: Let A_0 and S_0 denote the visible airplanes and satellites for the users, and F_0 denote the generated tasks in the given period. By utilizing interior point method to solve $\mathcal{P}2$ and $\mathcal{P}3'$ respectively, the computational complexity of can be calculated by $\mathcal{O}\left(l\cdot\left\{\left[F_0(A_0+S_0)\right]^3+\left[F_0(6S_0+2A_0+2)\right]^3\right\}\right)$.

As for the space complexity, it mainly depends on the sum of the optimality and feasibility cuts l adding to $\mathcal{P}3$. Also, each cut follows the store of all the optimization variables $\mathbf{p}$ and $\boldsymbol{\phi}$. Therefore, the space complexity can be obtained by $\mathcal{O}\left(l\cdot\max\left\{F_0(A_0+S_0), F_0(6S_0+2A_0+2)\right\}\right) = \mathcal{O}\left(l\cdot F_0(6S_0+2A_0+2)\right)$.

VI. NUMERICAL SIMULATIONS

In this section, the effectiveness of our work is verified via extensive numerical simulations. First, we analyzed the influence of transmit power and bandwidth on the outage probability of user-to-SAP links. Then, the convergence and optimality of the proposed MPTO algorithm are evaluated. Also, we evaluate the impact of specific parameters on the system performance.

A. Simulation Setup

Consider a scenario where 10 IoT devices generate computation-intensive tasks randomly on the ground. These users are distributed in the area of Donghai, and they generate 30 tasks every minute on average [33]. The investigated time is from 24 March 2021 04:00:00 UTCG to 25 March 2021 04:00:00 UTCG, and the TS is set as 1 s. There are 72 LEO satellites distributed over eight circular orbits at 600 km with the Walker constellation, and the locations of their corresponding GSs are at Leshan, Korla, Yantai and Shaoxing. The airplanes at the height of 10 km are considered to cover the target location 70% of the time. The error tolerance for the algorithm is set as $\epsilon_{\text{GBD}} = 10^{-3}$ and $\epsilon_f = 10^{-2}$. Other simulation parameters are described in Table III unless stated otherwise.

B. Feasibility of Transmission in CAA-SAGIN With Multi-Tier Computing Units

The minimum elevation of user-to-SAP is set as 10° [19]. Based on the geometric relationship, the average and maximum distance between users and CAs is 21.24 and 56.21 km. Also, the average and maximum distance between users and satellites are 1125.96 and 1931.64 km.

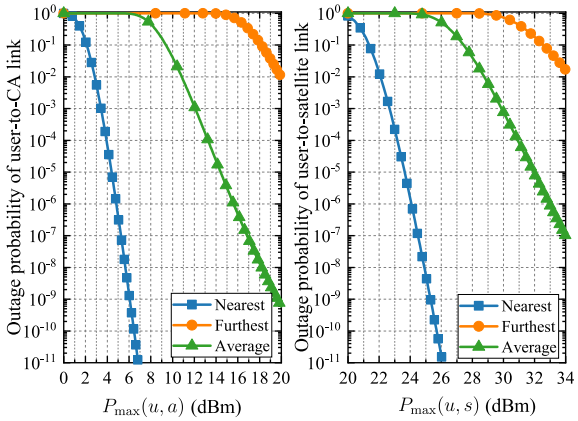


Fig. 3. Outage probability of user-to-SAP link v.s. Maximum transmit power $P_{\max}(u,i)$.

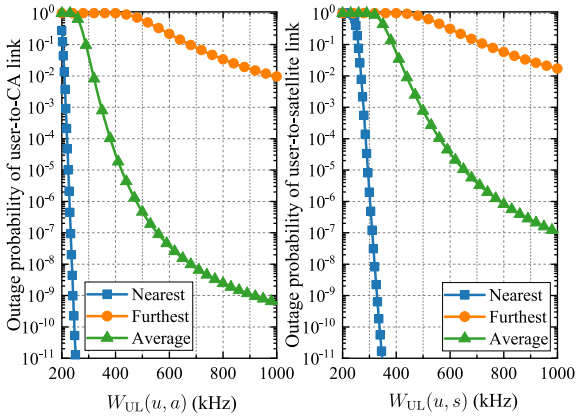


Fig. 4. Outage probability of user-to-SAP link v.s. Bandwidth $W_{\text{UL}}(u,i)$.

In order to reduce the transmission time between users and SAPs and ensure the effectiveness of computing by other platforms, we set the threshold value $R_{\text{th},f} = 2$ Mbps. Fig. 3 illustrates the impact of the user's maximum transmit power on the user-to-SAP link. The outage probability increases because the distance between the user and SAP increases due to more extensive path loss, and users need more power when transmitting to the satellite than to CA to maintain a similar outage probability for the same reason. The success rate remains over 99% after exceeding 10.83 dBm and 28.75 dBm under the average distance of AN and SN. Almost all the curves decrease to around 1% when $P_{\max}(u,i)$ exceeds 20 dBm and 34 dBm in AN and SN. It should be noticed that the maximum value is within the range of the IoT device's launch capability. Fig. 4 shows the relationship between outage probability and occupied bandwidth, which presents a similar trend. The outage probability of the blue lines (under nearest distance) falls below 10^{-11} after $P_{\max}(u,i)$ and $W_{\text{UL}}(u,i)$ exceed specific values. We cut the blue lines for readability when the outage probability is lower than 10^{-11} since these values can be almost negligible.

Aiming at guaranteeing the success rate of communication, we set $P_{\max}(u,a) = 100$ mW, $P_{\max}(u,s) = 2.5$ W and $W_{\text{UL}}(u,i) = 1$ MHz.

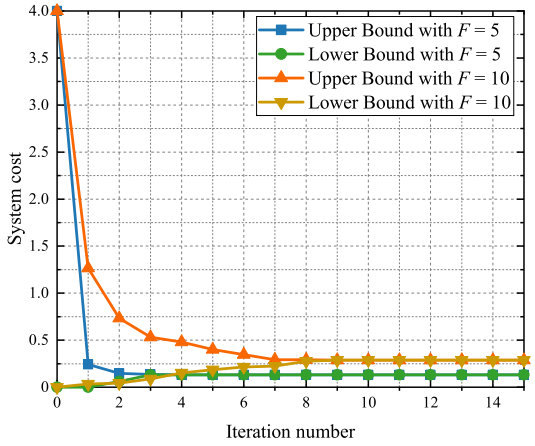


Fig. 5. Upper and lower bounds of outer iteration algorithm.

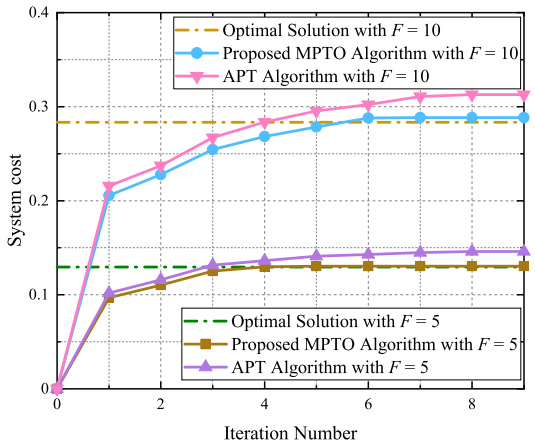
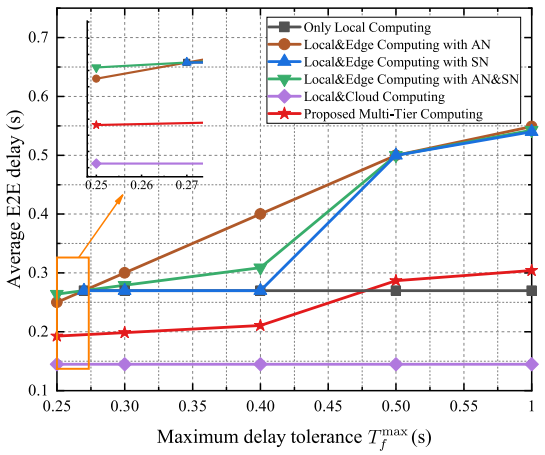
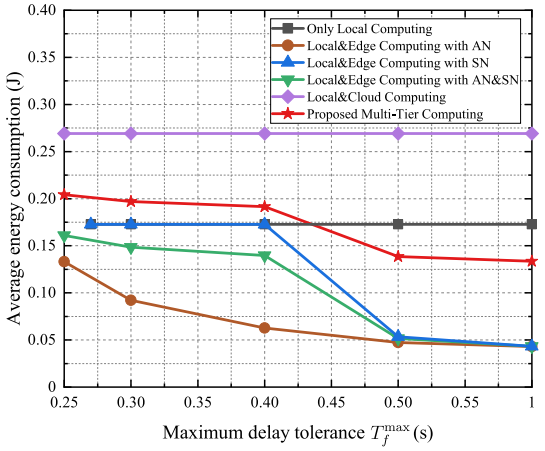


Fig. 6. Convergence analysis of inner iteration algorithm.



(a) Delay v.s. Maximum delay tolerance.



(b) Energy consumption v.s. Maximum delay tolerance.

Fig. 7. Effect of delay tolerance on system performance under different scenarios.

C. Convergence and Optimality of MPTO Algorithm

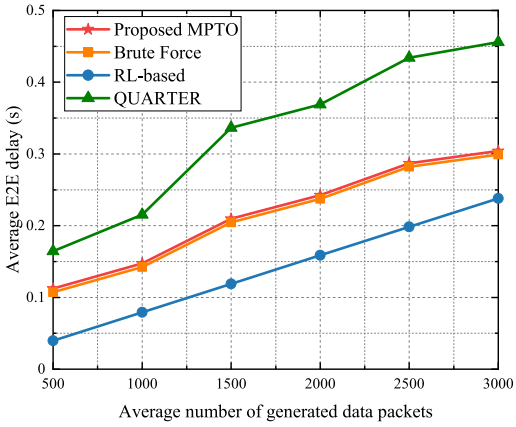
Fig. 5 verifies the convergence of the outer GBD method of the MPTO algorithm. We select two scenarios with 5 and 10 active users, each generating one task. Balancing factor η is set as 0.5, and the packets generated per user are 2500. It shows that the upper and lower bounds gradually approach each other and reach convergent values after limited (4 and 9) iterations with $F = 5$ and $F = 10$. The iteration speed decreases when the number of active users increases due to more optimization variables.

Recall that the inner iteration of the proposed MPTO algorithm adopts the modified parallel SCA algorithm. The optimality of the inner parallel SCA algorithm is presented in Fig. 6. The adaptive period of transmission (APT) algorithm [51] is selected as the baseline. The main idea of APT is to divide the original problem into several subproblems and find the optimal solution of the specific variable in each subproblem by fixing other ones until convergence. It can be found that the utilized parallel SCA algorithm approaches the optimal solution after several iterations, and its convergence speed is faster than that of APT. Due to the decoupling of many variables to be optimized, the system cost under APT surpasses the optimal solution. In contrast, the gap between the SCA algorithm and the optimal one is much closer since the SCA algorithm adopts the strategy of multi-variable joint optimization. These results also prove that selecting the final convergence value of the parallel SCA algorithm as the lower bound is rational.

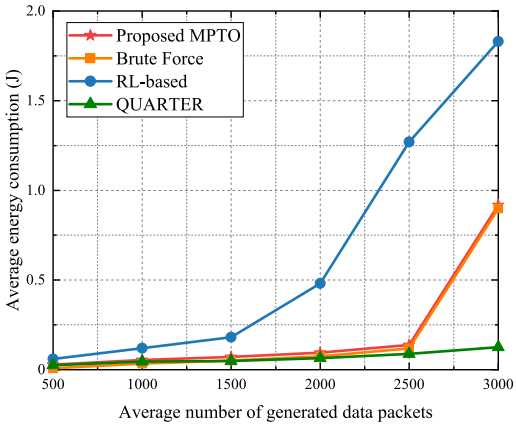
D. Performance Comparison of Multi-Tier Computing and Other Schemes

Fig. 7 illustrates the performance comparisons between the proposed multi-tier computing method and other five offloading schemes, namely 1) Only Local Computing, 2) Local&Edge Computing with AN, 3) Local&Edge Computing with SN, 4) Local&Edge Computing with AN&SN, 5) Local&Cloud Computing. We focus on the impact of computation-aware tasks' maximum delay tolerance $T^{\max}$ on system performance, and $T^{\max}$ varies in the range of [0.25, 1] s. The average number of generated packets is set as 2500.

The local computing scheme's E2E delay and energy consumption remain stable. The reason is that its objective function is a convex function, which is monotonously decreasing in the domain because $N_{\max}(u)$ is less than the KKT point. The curves of "Local&Cloud Computing" also remain stable because the cloud servers have huge computing capacity. The partial offloading between local and remote servers can achieve much lower latency without being limited by the delay tolerance.

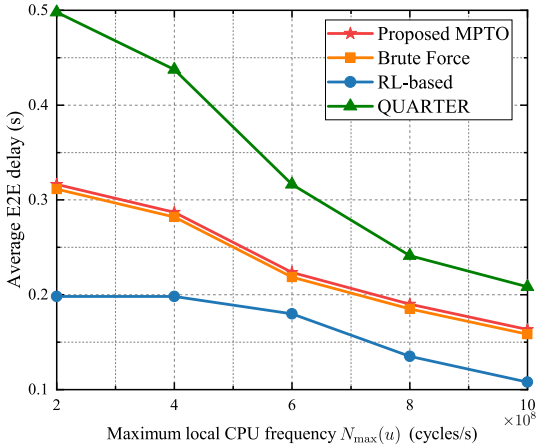


(a) Delay v.s. Packet number.

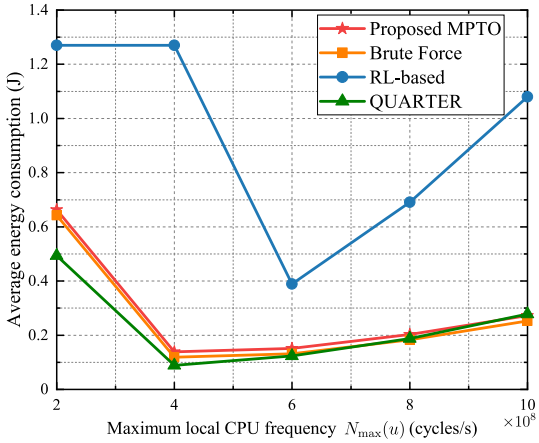


(b) Energy consumption v.s. Packet number.

Fig. 8. Effect of packet number on system performance under different algorithms.



(a) Delay v.s. Local CPU frequency.



(b) Energy consumption v.s. Local CPU frequency.

Fig. 9. Effect of local CPU frequency on system performance under different algorithms.

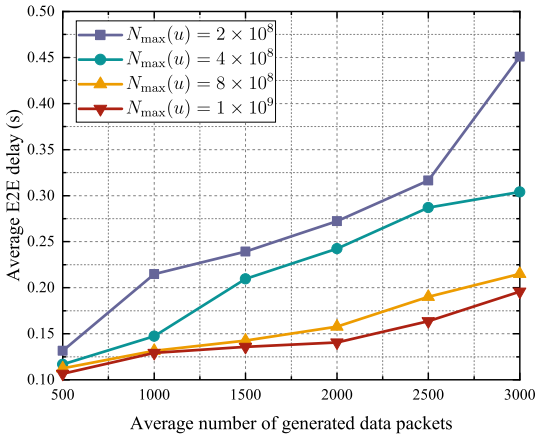
In Fig. 7(a), the delay of all the schemes is lower than $T^{\max}$ due to the delay constraints of the optimization problem. It can be noticed that the minimum delay under "Only Local Computing" and "Local&Edge Computing with SN" is 0.27 s, which is just the case that all the tasks are processed locally. Meanwhile, all the other schemes can achieve the minimum setting value of 0.25 s, and "Proposed Multi-Tier Computing" and "Local&Cloud Computing" schemes perform better than "Only Local Computing" when the tasks have strict delay requirements. All the lines except "Only Local Computing" and "Local&Cloud Computing" increase first and gradually becomes stable. Similarly, all the energy consumption lines except these two schemes decrease first and then become steady in Fig. 7(b). It can be explained that the required CPUs reduce due to the relaxation of delay tolerance. Also, it verifies the tradeoff relationship between delay and energy consumption. Furthermore, when the carbon emission of satellites can be ignored due to less influence on the earth, the energy consumption of satellites can be omitted either. In this case, the energy consumption curves related to SN will be even lower. The above facts prove the rationality and benefits of the proposed multi-tier computing scheme in CAA-SAGIN.

E. Performance Comparison of the Proposed MPTO Algorithm and Other Algorithms

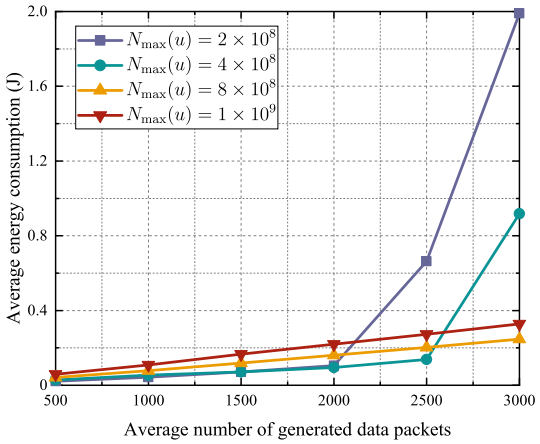
To show the efficiency of the proposed MPTO algorithm, we compare it with three typical existing computation offloading strategies in SAGIN, which are described as follows.

1) Brute Force: This strategy aims to solve the delay-energy optimization problem by comparing each offloading access platform and strategy. While achieving accurate results, its computation complexity is also very high.
2) RL-based [37]: This paper proposes a joint resource allocation and task scheduling scheme to minimize the E2E delay. A deep reinforcement learning (RL)-based computing offloading approach is utilized to obtain the optimal binary offloading strategy, where the tasks can be executed at the local device, edge servers (i.e., UAVs), or cloud servers (i.e., satellite gateway).
3) QUARTER [36]: This strategy minimizes average energy consumption under a partial offloading scheme while considering the queuing delay. The decomposed three subproblems are solved by the proposed QUARTER.

Fig. 8 shows the impact of packet number on system performance. With the increase of data packets, the average E2E delay and energy consumption increase accordingly. As expected, 'RL-based' reaches the lowest E2E delay, and 'QUARTER' has minimal energy consumption. In detail, aiming to minimize the energy consumption, 'QUARTER' prefers to execute locally or at edge servers rather than at remote servers. Therefore, when there are more data packets,

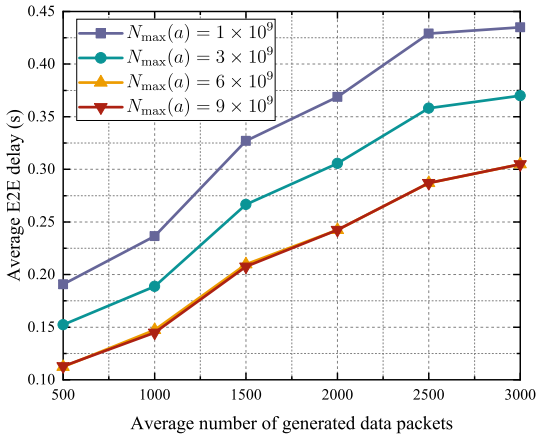


(a) Delay v.s. Packet number.

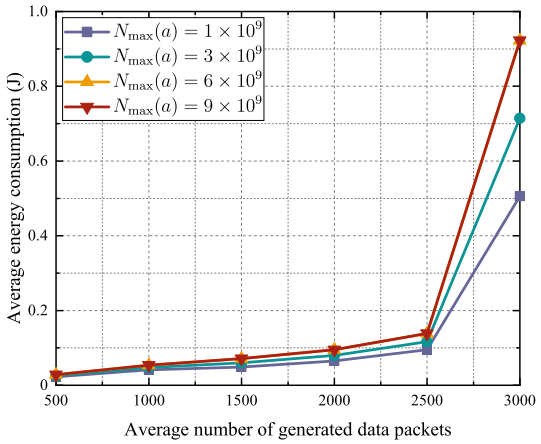


(b) Energy consumption v.s. Packet number.

Fig. 10. Effect of packet number on system performance under different local CPU frequency.



(a) Delay v.s. Packet number.



(b) Energy consumption v.s. Packet number.

Fig. 11. Effect of packet number on system performance under different CA's CPU frequency.

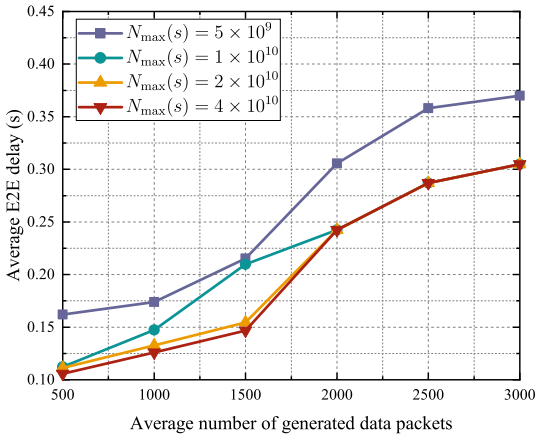
its average E2E delay and queuing delay become higher than the other three schemes. As for 'RL-based' aiming to minimize the total delay, the tasks prefer to be offloaded to remote servers while satisfying the delay tolerance. Since there is an exponential increase in energy consumption as the processed data volume increases, the energy consumption under 'RL-based' is much higher than the other three schemes. To jointly minimize E2E delay and energy consumption, 'Proposed MPTO' and 'Brute Force' can balance these two objectives well. Meanwhile, 'Proposed MPTO' performs very close to 'Brute Force', representing the effectiveness of 'Proposed MPTO'.

We also compare these four schemes by varying the maximum local CPU frequency in the range of $[2, 8] \times 10^8$ in Fig. 9. With the increase of local computing capacity, the E2E delay shows a downward trend in Fig. 9(a), while the energy consumption decreases first and then increases 9(b). In detail, when the local CPU frequency is lower, the delay and energy consumption of 'RL-based' remain stable because the tasks prefer to be offloaded to other platforms for processing to satisfy the strict delay tolerance. For the other three partial offloading schemes, the E2E delay keeps falling because of the increase in local computing capability. In terms of energy consumption, improving local CPU frequency increases the processing ratio at IoT devices, which reduces the energy consumption of transmitting to SAPs and computing at other platforms at first. However, after exceeding the threshold of local CPU frequency, the increasing percentage of local processing performs worse than offloading due to the more significant local computing energy consumption. Thus, the offloaded ratio increases, and the inflection points of energy consumption appear.

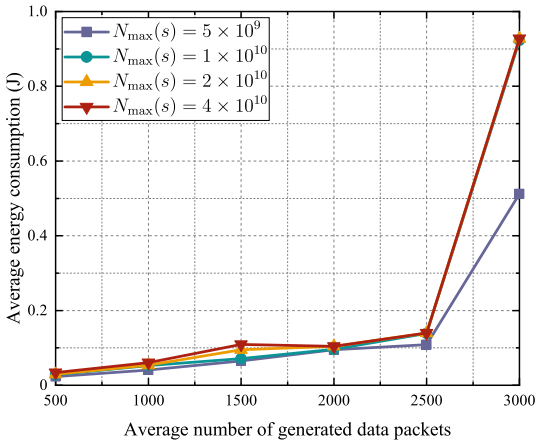
F. Effect of Computation Capacity on System Performance

In order to verify the impact of packet amount on system performance, Fig. 10 compares the system performance under 'Proposed MPTO' scheme with different local computation capacity $N_{\max}(u)$. When the generated packets are fewer, the scenarios with larger $N_{\max}(u)$ have lower delay and higher energy consumption as expected. However, when the packet amount exceeds specific values, the energy consumption with lower $N_{\max}(u)$ exceeds those with higher one. This can be explained by the fact that the IoT devices cannot support long-packet computation tasks locally and need other platforms for efficient parallel computing to ensure the delay requirements. Thus, energy consumption rises sharply.

To investigate the impact of SAP's maximal CPU frequency on system performance, we plot Fig. 11 and Fig. 12 by



(a) Delay v.s. Packet number.



(b) Energy consumption v.s. Packet number.

Fig. 12. Effect of packet number on system performance under different satellite's CPU frequency.

varying CA's and satellite's computation capacity. Specifically, when $N_{\max}(a)$ increases, the average E2E delay decreases first and then becomes stable, while the energy consumption increases first and remains unchanged. Similarly, with the increase of satellite computation capacity, the delay and energy consumption can also be stabilized after $N_{\max}(s)$ is over a specific value. The objective of joint delay-energy optimization can account for these facts. Although improving SAP processing capability can bring out delay benefits, more energy consumption at edge servers will also increase the system cost. In this case, the proposed MPTO algorithm will make the offloading strategy intelligently and determine the proper processing ratio at different platforms.

Fig. 10 – Fig. 12 also illustrate that better system performance cannot be achieved only by improving devices' computing capability. This trend illustrates that the complex coupling relationships between multiple variables should be considered in the multi-tier hybrid offloading architecture.

VII. Conclusion

This paper investigated the multi-tier computation offloading problem in dynamic CAA-SAGIN, aiming to minimize the weighted sum of delay and energy consumption. To fully exploit the advantages of different platforms, we considered a local-edge-cloud hybrid offloading scheme, making the issue more comprehensive and challenging due to the inner- and inter-coupling interactions of communication and computing resources. To overcome the problem of the high complexity of the original MINLP problem, we decomposed it into two subproblems, i.e., primal and master problems, which achieve upper and lower bounds respectively at each iteration. The modified parallel SCA algorithm converted the NP-hard master problem into a tractable convex one. The proposed MPTO algorithm obtained the optimization solutions with association strategy, transmit power, computation frequency, task partition and offloading ratio. Simulation results verified the convergence and optimality of the MPTO algorithm. Compared to existing offloading algorithms in SAGIN, the proposed MPTO algorithm can reach the optimal tradeoff between E2E delay and energy consumption when satisfying a more strict delay tolerance.

This work can be regarded as the initial attempt in the multi-tier hybrid computation offloading of CAA-SAGIN. There is still a lot of work worth pursuing in the future. For example, when considering the task queue at the entities with processing capability, it is vital to arrange the processing order of the tasks with different priorities while guaranteeing the stability of the task queue and realizing the optimization objective.

Appendix A
Proof of the Proposition 1

When $x_f(u,s) = 1$, there are two extreme cases. The first one is all the offloaded data is computed by the accessed satellite, and the other one is all tasks are transmitted to the GS for processing. The first case is the same as when user accesses to AN for task offloading, so we only discuss the case of uploading tasks to satellites. First, we can deduce $\rho_f \geq 1 - \frac{z_{f,1}}{b_f}$ from (18f). Then, the following expressions can be obtained with (18h) and (18l) for the first case,

$$\begin{aligned}\rho_f &\geq 1 - \frac{\frac{T_f^{\max}}{x_f(u,s)} - T_f^{\text{SN,pro,1}}(u,s)}{b_f R_p \left[\frac{1}{R_f(u,s)} + \frac{\omega_s}{n_f^{\text{EC}}(s)} + \frac{\alpha}{R_f(s,u)}\right]} \\ &\geq 1 - \frac{\frac{T_f^{\max}}{x_f(u,s)} - T_f^{\text{SN,pro,1}}(u,s)}{b_f R_p \left[\frac{1}{R_f(u,s)} + \frac{\omega_s}{N_{\max}(s)} + \frac{\alpha}{R_f(s,u)}\right]}. \end{aligned} \quad (37)$$

For the second case, with (18l), we have

$$\rho_f \geq 1 - \frac{\frac{T_f^{\max}}{x_f(u,s)} - T_f^{\text{SN,pro,2}}(u,s)}{R_p b_f \left(\frac{1}{R_f(u,s)} + C_{f,1}(u,s)\right)}. \quad (38)$$

Then, (19) can be obtained.

References

[1] *The Mobile Economy 2022*. Accessed: Mar. 12, 2022. [Online]. Available: https://www.gsma.com/mobileeconomy/

[2] S. S. Hajam and S. A. Sofi, "IoT-fog architectures in smart city applications: A survey," *China Commun.*, vol. 18, no. 11, pp. 117–140, Nov. 2021.

[3] D. You, B.-S. Seo, E. Jeong, and D. H. Kim, "Internet of Things (IoT) for seamless virtual reality space: Challenges and perspectives," *IEEE Access*, vol. 6, pp. 40439–40449, 2018.

[4] C. Neff, M. Mendieta, S. Mohan, M. Baharani, S. Rogers, and H. Tabkhi, "REVAMP2T: Real-time edge video analytics for multicamera privacy-aware pedestrian tracking," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2591–2602, Apr. 2020.

[5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generat. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Feb. 2013.

[6] Y. Yang, "Multi-tier computing networks for intelligent IoT," *Nature Electron.*, vol. 2, no. 1, pp. 4–5, Jan. 2019.

[7] V. Cardellini et al., "A game-theoretic approach to computation offloading in mobile cloud computing," *Math. Program.*, vol. 157, no. 2, pp. 421–449, Jun. 2016.

[8] J. Ren, G. Yu, Y. He, and G. Y. Li, "Collaborative cloud and edge computing for latency minimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5031–5044, May 2019.

[9] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4804–4814, Jun. 2019.

[10] M. Sheng, Y. Wang, X. Wang, and J. Li, "Energy-efficient multiuser partial computation offloading with collaboration of terminals, radio access network, and edge server," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1524–1537, Mar. 2020.

[11] Q. Zhang, L. Gui, F. Hou, J. Chen, S. Zhu, and F. Tian, "Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud RAN," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3282–3299, Apr. 2020.

[12] X. Chen, Y. Zhou, L. Yang, and L. Lv, "User satisfaction oriented resource allocation for fog computing: A mixed-task paradigm," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6470–6482, Oct. 2020.

[13] L. Feng, W. Li, Y. Lin, L. Zhu, S. Guo, and Z. Zhen, "Joint computation offloading and URLLC resource allocation for collaborative MEC assisted cellular-V2X networks," *IEEE Access*, vol. 8, pp. 24914–24926, 2020.

[14] F. Song, H. Xing, S. Luo, D. Zhan, P. Dai, and R. Qu, "A multiobjective computation offloading algorithm for mobile-edge computing," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8780–8799, Sep. 2020.

[15] J. Zhang, W. Xia, F. Yan, and L. Shen, "Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing," *IEEE Access*, vol. 6, pp. 19324–19337, 2018.

[16] (Jun. 8, 2020). *SpaceX: We've Launched 32,000 Linux Computers Into Space for Starlink Internet*. [Online]. Available: https://www.zdnet.com/article/spacex-weve-launched-32000-linux-computers-into-space-for-starlink-internet/

[17] S. Li, Q. Chen, Z. Li, W. Meng, and C. Li, "Civil aircraft assisted space-air-ground integrated networks: Architecture design and coverage analysis," *China Commun.*, vol. 19, no. 1, pp. 29–39, Jan. 2022.

[18] Q. Chen, S. Li, W. Meng, and C. Li, "Capacity analysis of civil aircraft networks in SAGIN," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Seoul, South Korea, May 2022, pp. 1877–1882.

[19] Q. Chen, W. Meng, S. Han, C. Li, and H.-H. Chen, "Robust task scheduling for delay-aware IoT applications in civil aircraft-augmented SAGIN," *IEEE Trans. Commun.*, vol. 70, no. 8, pp. 5368–5385, Aug. 2022.

[20] Q. Chen, W. Meng, S. Li, C. Li, and H.-H. Chen, "Civil aircrafts augmented space–air–ground-integrated vehicular networks: Motivation, breakthrough, and challenges," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5670–5683, Apr. 2022.

[21] *The Connected Aircraft: How Aviation is Changing*. Accessed: Mar. 12, 2022. [Online]. Available: https://aerospace.honeywell.com/content/dam/aerobt/en/documents/learn/challenges/infographics/HON_Infographic_ConnectedAircraft_2019_June25.pdf

[22] C. F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4132–4150, Jun. 2019.

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

[24] Z. Zhang, W. Zhang, and F.-H. Tseng, "Satellite mobile edge computing: Improving QoS of high-speed satellite-terrestrial networks using edge computing techniques," *IEEE Netw.*, vol. 33, no. 1, pp. 70–76, Jan. 2019.

[25] R. Xie, Q. Tang, Q. Wang, X. Liu, F. R. Yu, and T. Huang, "Satellite-terrestrial integrated edge computing networks: Architecture, challenges, and open issues," *IEEE Netw.*, vol. 34, no. 3, pp. 224–231, May 2020.

[26] Z. Hu, F. Zeng, Z. Xiao, B. Fu, H. Jiang, and H. Chen, "Computation efficiency maximization and QoE-provisioning in UAV-enabled MEC communication systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1630–1645, Apr. 2021.

[27] Y. Wang, J. Yang, X. Guo, and Z. Qu, "A game-theoretic approach to computation offloading in satellite edge computing," *IEEE Access*, vol. 8, pp. 12510–12520, 2020.

[28] Z. Song, Y. Hao, Y. Liu, and X. Sun, "Energy-efficient multiaccess edge computing for terrestrial-satellite Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 14202–14218, Sep. 2021.

[29] G. Cui, X. Li, L. Xu, and W. Wang, "Latency and energy optimization for MEC enhanced SAT-IoT networks," *IEEE Access*, vol. 8, pp. 55915–55926, 2020.

[30] R. Duan, J. Wang, C. Jiang, Y. Ren, and L. Hanzo, "The transmit-energy vs computation-delay trade-off in gateway-selection for heterogenous cloud aided multi-UAV systems," *IEEE Trans. Commun.*, vol. 67, no. 4, pp. 3026–3039, Apr. 2019.

[31] S. Yu, X. Gong, Q. Shi, X. Wang, and X. Chen, "EC-SAGINs: Edge-computing-enhanced space–air–ground-integrated networks for internet of vehicles," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5742–5754, Apr. 2022.

[32] B. Cao et al., "Edge–cloud resource scheduling in space–air–ground-integrated networks for internet of vehicles," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5765–5772, Apr. 2022.

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

[34] C. Zhou et al., "Delay-aware IoT task scheduling in space-air-ground integrated network," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[35] S. Zhang, G. Cui, Y. Long, and W. Wang, "Joint computing and communication resource allocation for satellite communication networks with edge computing," *China Commun.*, vol. 18, no. 7, pp. 236–252, Jul. 2021.

[36] H. Liao, Z. Zhou, X. Zhao, and Y. Wang, "Learning-based queue-aware task offloading and resource allocation for space–air–ground-integrated power IoT," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5250–5263, Apr. 2021.

[37] N. Cheng et al., "Space/aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.

[38] Z. Ning et al., "Dynamic computation offloading and server deployment for UAV-enabled multi-access edge computing," *IEEE Trans. Mobile Comput.*, early access, Nov. 23, 2021, doi: 10.1109/TMC.2021.3129785.

[39] B. Di, H. Zhang, L. Song, Y. Li, and G. Y. Li, "Ultra-dense LEO: Integrating terrestrial-satellite networks into 5G and beyond for data offloading," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 47–62, Jan. 2019.

[40] C. You and R. Zhang, "3D trajectory optimization in Rician fading for UAV-enabled data harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3192–3207, Jun. 2019.

[41] O. Amin, E. Bedeer, M. H. Ahmed, and O. A. Dobre, "Energy efficiency-spectral efficiency tradeoff: A multiobjective optimization approach," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 1975–1981, Apr. 2016.

[42] A. M. Geoffrion, "Generalized Benders decomposition," *J. Optim. Theory Appl.*, vol. 10, no. 4, pp. 237–260, Apr. 1972.

[43] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, Jul. 2009.

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

[45] G. Scutari and Y. Sun, "Parallel and distributed successive convex approximation methods for big-data optimization," in *Multi-Agent Optimization* (Lecture Notes in Mathematics), vol. 2224. USA: Springer-Verlag, Nov. 2018, pp. 141–308.

[46] J. Tang, D. K. C. So, E. Alsusa, and K. A. Hamdi, "Resource efficiency: A new paradigm on energy efficiency and spectral efficiency tradeoff," *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4656–4669, Aug. 2014.

[47] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1746–1755, Sep. 2004.

[48] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proc. USENIX HotCloud*, Boston, MA, USA, Jun. 2010, pp. 4–11.
[49] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Joint allocation of computation and communication resources in multiuser mobile cloud computing," in *Proc. IEEE 14th Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Darmstadt, Germany, Jun. 2013, pp. 26–30.
[50] D. Zhou, M. Sheng, X. Wang, C. Xu, R. Liu, and J. Li, "Mission aware contact plan design in resource-limited small satellite networks," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2451–2466, Jun. 2017.
[51] S. Fu, J. Gao, and L. Zhao, "Integrated resource management for terrestrial-satellite systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3256–3266, Mar. 2020.

Qian Chen (Graduate Student Member, IEEE) received the B.E. degree in electronic and information engineering from the Harbin Institute of Technology (HIT), Harbin, China, in 2018, where she is currently pursuing the Ph.D. degree with the Communications Research Center. She has been a Visiting Student with the Information Systems Technology and Design (ISTD), Singapore University of Technology and Design (SUTD), Singapore, since December 2021. Her current research interests include resource allocation and mobility management in space-air-ground integrated networks. She is the Student Chair of IEEE ComSoc Harbin Chapter. She has served as a TPC Member for GLOBECOM 2021 and ICC 2022.

Weixiao Meng (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 1990, 1995, and 2000, respectively.

From 1998 to 1999, he worked at NTT DoCoMo, as a Visiting Researcher, and the University of California at Riverside as a Senior Visiting Scholar in 2017. He is currently a Full Professor with the School of Electronics and Information Engineering, HIT. He is also the Director of on China-Chile ICT Belt and Road Joint Laboratory, the Deputy Director of the Engineering Research Center of Dedicate Communication System, Ministry of Education (MOE), China. His research interests include broadband wireless communications, space-air-ground integrated networks, and wireless localization technologies. He is a fellow of the China Institute of Electronics and a Senior Member of the IEEE ComSoc and the China Institute of Communication. Under his leading, Harbin Chapter won IEEE ComSoc Chapter of the Year Award, the Asia Pacific Region Chapter Achievement Award, and the Personal Member and Global Activities Contribution Award in 2018. In 2021, he won the Best Paper Award of IEEE SYSTEMS JOURNAL. From 2020 to 2022, he was selected into the Top 2% of the World's Scientists by Stanford University. He is the Chair of IEEE ComSoc Harbin Chapter. He acted as the leading TPC Co-Chair of ChinaCom 2011 and ChinaCom 2016, the leading Services and Applications Track Co-Chair of WCNC 2013, the Awards Co-Chair of ICC 2015 and the Wireless Networking Symposia Co-Chair of GLOBECOM 2015, the AHSN Symposia Co-Chair of GLOBECOM 2018 and ICC 2020, and the leading Workshop Co-Chair of ICC 2019 and ICNC 2020. In 2005, he was an Honored Provincial Excellent Returnee and selected into New Century Excellent Talents (NCET) plan by MOE, China, in 2008, and the Distinguished Academic Leader of Harbin in 2015. He has been an Editorial Board Member for Wiley's WCMC Journal from 2010 to 2017, an Area Editor for *PHYCOM* journal from 2014 to 2016, and an editorial board for IEEE COMMUNICATIONS SURVEYS AND TUTORIALS from 2014 to 2017, and IEEE WIRELESS COMMUNICATIONS since 2015.

Tony Q. S. Quek (Fellow, IEEE) received the B.E. and M.E. degrees in electrical and electronics engineering from the Tokyo Institute of Technology in 1998 and 2000, respectively, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology in 2008.

He is currently the Cheng Tsang Man Chair Professor with Singapore University of Technology and Design (SUTD). He also serves as the Director of the Future Communications Research and Development Program, the Head of ISTD Pillar, and the Deputy Director of the SUTD-ZJU IDEA. His current research topics include wireless communications and networking, network intelligence, the Internet of Things, URLLC, and 6G.

Dr. Quek is a fellow of the Academy of Engineering Singapore. He was an Executive Editorial Committee Member for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He has been actively involved in organizing and chairing sessions and also served as a member of the Technical Program Committee as well as the symposium chairs in a number of international conferences. He was honored with the 2008 Philip Yeo Prize for Outstanding Achievement in Research, the 2012 IEEE William R. Bennett Prize, the 2015 SUTD Outstanding Education Awards–Excellence in Research, the 2016 IEEE Signal Processing Society Young Author Best Paper Award, the 2017 CTTC Early Achievement Award, the 2017 IEEE ComSoc AP Outstanding Paper Award, the 2020 IEEE Communications Society Young Author Best Paper Award, the 2020 IEEE Stephen O. Rice Prize, the 2020 Nokia Visiting Professor, and the 2016–2020 Clarivate Analytics Highly Cited Researcher. He is currently serving as an Area Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and an Elected Member of the IEEE Signal Processing Society SPCOM Technical Committee. He was an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and the IEEE WIRELESS COMMUNICATIONS LETTERS.

Shuyi Chen (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in information and communication engineering from the School of Electronics and Information Engineering, Harbin Institute of Technology (HIT), Harbin, China, in 2013, 2015, and 2021, respectively. She is currently an Assistant Professor with HIT. Her research interests include wireless communication and networking technologies, performance analysis and interference management in ultra-dense networks, and application of artificial intelligence in future wireless networks.