



OPEN Phase shift optimization in reconfigurable intelligent surface-assisted UAV in hierarchical aerial computing networks

Basma Diao[✉], Ibrahim I. Ibrahim, Ahmed M. Abdelhaleem & Mostafa M. Abdelhakam

The evolution toward 6G wireless networks necessitates innovative solutions to support the massive Internet of Things (IoT) deployments with unprecedented computational and communication requirements, motivating this A comprehensive framework that integrates Reconfigurable Intelligent Surface (RIS) technology with hierarchical aerial computing networks by combining RIS-equipped Unmanned Aerial Vehicles (UAVs) operating as mobile edge computing nodes with High-Altitude Platforms (HAPs) to create a three-tier computing hierarchy addressing the limitations of conventional terrestrial infrastructure. The system model encompasses RIS-equipped UAVs serving terrestrial IoT devices with a single HAP providing high-capacity computational resources, where the RIS phase optimization is formulated as a Riemannian conjugate gradient problem on complex circle manifolds to maximize total system throughput while naturally handling unit modulus constraints through a three-stage sequential decomposition approach. Extensive Monte Carlo simulations demonstrate significant performance improvements over the comparable algorithm without RIS enhancement, with the RIS-enhanced system achieving 18% throughput improvement, near-linear scalability serving approximately 100% of available IoT devices compared to the algorithm. In the comparable algorithm at 100 devices, a 95% task completion rate was maintained across all network loads versus 79–80% for the algorithm compared to the comparable algorithm. The results validate the potential of RIS-enabled aerial networks as a transformative solution for scalable and efficient 6G IoT services, with enhanced channel quality from intelligent phase configuration, enabling superior resource utilization and service provisioning in hierarchical computing architectures, establishing key contributions including novel RIS-aerial computing integration, advanced Riemannian manifold optimization with superior convergence properties, unified resource allocation combining stable matching theory with externality elimination, comprehensive performance analysis demonstrating practical viability, and real-world implementation considerations for future multi-UAV scenarios and energy-efficient designs.

Keywords Aerial computing, unmanned aerial vehicle (UAV), High altitude platform (HAP), Reconfigurable intelligent surfaces (RIS), Mobile edge computing (MEC), Resource allocation, Matching game theory, Riemannian conjugate gradient (RCG)

The evolution toward 6G wireless networks necessitates innovative solutions to support massive IoT deployments with stringent quality of service requirements^{1,2}. Recent advances in Reconfigurable Intelligent Surfaces (RIS) offer unprecedented opportunities to enhance wireless links through intelligent signal manipulation^{3,4}. While hierarchical aerial computing networks using HAP and UAV have shown promise for providing mobile edge computing services⁵, their communication performance remains fundamentally limited by adverse channel conditions and interference, particularly in dynamic three-dimensional service topologies. However, integrating RIS with multi-tier aerial computing platforms presents unique and largely unaddressed challenges. These challenges stem from the complex coupling between RIS phase configuration, dynamic resource allocation, and the hierarchical offloading decisions across IoT, UAV, and HAP tiers.

Existing literature has made significant progress in two distinct domains. On one hand, works like⁵ have developed sophisticated resource allocation and matching algorithms for HAP-UAV-IoT hierarchies, but without considering RIS for channel enhancement. On the other hand, RIS optimization has been extensively studied

Department of Electronics and Communications Engineering, Faculty of Engineering, Helwan University, Cairo, Egypt. ✉email: basma.diao@h-eng.helwan.edu.eg

for simple, static, or single-tier communication scenarios^{4,6}. A critical research gap exists in jointly optimizing the RIS phase configuration within a dynamic, three-tier (IoT-UAV-HAP) computation offloading framework. Prior approaches typically treat RIS beamforming and hierarchical task scheduling as separate problems. This decoupling fails to capture the essential system dynamics: the optimal RIS phase shifts for IoT-UAV links directly influence the uplink data rate, which in turn determines the latency and feasibility of local UAV computation versus offloading to the HAP. Consequently, there is a lack of a unified optimization framework that: (1) natively and efficiently handles the unit modulus constraints of passive RIS elements, (2) coordinates stable device-association in the presence of resource competition and preference interdependencies, and (3) jointly makes intelligent computation placement decisions across the aerial hierarchy based on RIS-enhanced channel states. This paper bridges this gap by introducing a novel RIS-assisted hierarchical computing system. We address the joint optimization of RIS phase configuration and multi-tier resource allocation in an integrated model. The primary challenge lies in maximizing the total system throughput while ensuring stable matching, respecting individual energy budgets, and guaranteeing stringent end-to-end delay constraints for all tasks. To this end, we develop a hybrid algorithmic solution that decomposes the complex problem into manageable yet coordinated sub-problems.

Related work

Computation offloading has gained significant attention in recent years as a means to overcome the limited processing and energy resources of IoT devices. With the proliferation of IoT devices and the advent of 5G and beyond wireless technologies, traditional cloud computing approaches face significant challenges in meeting the stringent latency and bandwidth requirements of emerging applications. Aerial platforms, particularly UAVs and HAPs, have emerged as promising enablers for MEC due to their flexibility, wide coverage, and ability to complement terrestrial infrastructures. For example in⁷ proposed UAV-assisted MEC architectures where UAVs act as relays forwarding computation tasks from IoT devices to ground edge servers. The authors formulated a joint optimization problem involving IoT-UAV-ES associations and UAV deployment topology, developing a Restricted Three-Sided Matching with Size and Cyclic Preference (R-TMSC) algorithm. Hierarchical aerial computing architectures were further explored in⁵, Jia investigated joint trajectory and resource optimization in UAV-HAP MEC networks assisted by RIS, demonstrating significant throughput gains through tiered offloading. Similarly, Zhao et al.⁸ examined energy-latency trade-offs in RIS-assisted aerial computing, highlighting the role of intelligent surfaces in improving energy efficiency. Li et al.⁹ further expanded the scope to multi-HAP multi-UAV edge computing networks, incorporating mobility-aware dynamic resource allocation.

For Computation Offloading Methodologies¹⁰, cover fundamental elements of DRL algorithm design and highlighting superior perception and decision-making capabilities compared to traditional optimization. The integration of blockchain with MEC is explored in¹¹ which presents an A3C-AHP framework for multi-user optimization in blockchain-enabled IoT systems. For industrial IoT applications¹², addresses computation offloading in multi-server edge computing environments with joint load balancing and security considerations, incorporating fuzzy security mechanisms.

A comprehensive study on resource provisioning and optimization in edge computing is provided in¹³, which offers a holistic review of static, dynamic, and user-centric resource allocation models and examines the integration of Software Defined Networks (SDN) for enhancing resource orchestration. The survey highlights critical challenges, including scalability, interoperability, and security in managing dynamic and heterogeneous infrastructures.

RIS technology has emerged as a transformative solution for improving wireless communication by creating controllable electromagnetic environments^{14,15}. Recent advancements have focused on RIS integration with aerial platforms with studies exploring intelligent reflecting surfaces for multi-UAV NOMA communications and implementation using compressive sensing and deep learning techniques¹⁶. Huang et al.¹⁷ provide a comprehensive survey on RIS-assisted UAV communications, covering joint beamforming, trajectory design, and energy efficiency. Nguyen¹⁸ investigate joint UAV trajectory and RIS phase shift optimization for IoT data collection, while Wang¹⁹ apply deep reinforcement learning for secure RIS-assisted UAV communications in dynamic environments. In the context of phase optimization, Mei²⁰ advance Riemannian optimization techniques for RIS phase design in multi-user MIMO systems, offering methodological enhancements relevant to manifold-based approaches. Practical implementation aspects are addressed by Zhou²¹, who provide experimental validation of RIS-assisted UAV links in urban scenarios.

Contribution

The key contributions of this study are outlined in the following points:

- We propose the first comprehensive framework that integrates RIS technology with hierarchical aerial computing networks, combining RIS-equipped UAVs operating as mobile edge computing nodes with HAPs to create a three-tier computing hierarchy. Unlike existing works that consider either aerial computing or RIS enhancement separately, our integrated approach addresses both communication and computation challenges simultaneously, creating synergistic performance improvements beyond what either technology achieves independently.
- The RIS phase optimization is formulated as a Riemannian conjugate gradient problem on the complex circle manifolds, representing a significant methodological advancement over conventional optimization approaches. This geometric formulation naturally handles unit modulus constraints inherent in passive RIS elements while achieving superior convergence properties compared to penalty-based or alternating optimization methods commonly used in prior work. The manifold-based approach exploits the intrinsic geometric structure of the constraint space, enabling more efficient and stable optimization.

- We develop a three-stage sequential decomposition algorithm that uniquely combines (i) stable matching theory with externality elimination for IoT-UAV associations, addressing preference interdependencies that existing matching approaches ignore; (ii) Riemannian manifold optimization for RIS phase configuration; and (iii) intelligent heuristics for hierarchical UAV-HAP task distribution. Unlike prior hierarchical computing works that separate resource allocation from communication enhancement, our unified approach ensures that improved channel conditions from RIS optimization directly inform and enhance resource allocation decisions.
- We provide extensive simulation results demonstrating substantial improvements over comparable hierarchical aerial computing systems: 18% throughput improvement, near-linear scalability serving approximately 100% of available IoT devices (compared to plateauing at ~80% in baseline approaches), 95% task completion rate maintained across all network loads versus 79–80% for conventional systems, and 28.6% reduction in average task processing delay. These results establish the practical viability and quantifiable benefits of RIS-enhanced aerial computing.
- Unlike theoretical studies that assume idealized conditions, we address real-world constraints including finite energy budgets, strict delay requirements, practical RIS hardware limitations (amplitude losses, phase resolution), and computational complexity considerations. We also analyze the energy-performance tradeoffs inherent in RIS operation and provide insights for system designers on when and how to deploy RIS-enhanced aerial computing platforms.

The combination of these contributions establishes a new paradigm for scalable and efficient 6G IoT services, where intelligent manipulation of the wireless environment through RIS technology fundamentally enhances the capabilities of hierarchical aerial computing architectures.

The rest of this work is arranged as follows. A comprehensive system model is presented in Section II, including the hierarchical network architecture, RIS-enhanced channel models, time and energy cost models, and the mathematical problem formulation. The algorithms are proposed in Section III, encompassing the stable matching algorithm with externality elimination for IoT-UAV associations, the Riemannian conjugate gradient algorithm for RIS phase optimization and the heuristic approach for UAV-HAP task offloading. Section IV provides extensive simulation results and performance evaluation, comparing the proposed RIS-enhanced system against baseline approaches across various metrics, including total computed data, energy consumption, and task completion rates. conclusions are offered in Section V. Finally, Section VI concludes the papers and outlines potential research directions for RIS-enhanced aerial computing networks. Besides, for clarity, the notations used in this work are listed in Table 1.

Notation	Parameters	Notation	Parameters
\mathcal{U}	UAV set, $u \in \mathcal{U}$	K_i	Data size of IoT $i \in \mathcal{I}$.
\mathcal{I}	IoT user set, $i \in \mathcal{I}$.	D_i	IoT Delay Tolerance.
ρ_u, μ	Computation resource cost of UAV, HAP.	H_u	Flight altitude of UAV u .
C_u, \mathcal{C}	Computing capability of UAV, HAP.	σ^2	Noise Power Density.
q_u, q_i	Horizon location of UAV, IoT.	G_0	Reference Channel Gain.
N_u	The maximum number of IoT a UAV can serve.	τ_0	Initial Step.
T_{iu}	Time cost to transmit the data of I2U.	ρ	Backtracking Factor.
T_u	Time cost to transmit the data to HAP by UAV u .	c_1	Armijo Parameter
T_u^i	Time cost by UAV to complete the computation.	ϵ	Convergence tolerance
T^i	Time cost by HAP to complete the computation.	N	Reflecting elements
P_i^{tr}	Transmission power of I2U.	α_n	Amplitude Reflection Coefficients .
P_u^{tr}	Transmission power of U2H.	$\lambda_1, \lambda_2, \lambda_3$	Matching Weights.
ζ_u, ζ	Energy consumption coefficient of UAV, HAP.	w_1, w_2	Heuristic Weights .
E_i^c, E_u^c, E^c	Total energy cost of IoT, UAV, HAP.	$f_{u,h}$	U2H Center Frequency.
E_i^o, E_u^o, E^o	Basic operation energy cost of IoT, UAV, HAP.	T_s	System Noise Temp.
E_i, E_u, E	Energy budget of IoT, UAV, HAP.	k_B	Boltzmann Constant.
E_i^{tr}	Energy cost for I2U data transmission.	BW_{iu}	Bandwidth of I2U channel.
E_u^{tr}	Energy cost for U2H data transmission.	BW_u	Bandwidth of U2H channel.
E_u^{co}	Energy cost for computation of u .	$R_{i,u}$	Data rate of I2U channel.
E^{co}	Energy cost of HAP for computation.	R_u	Data rate of U2H channel.
M_1	Matching in Algorithm 1.	$A_u^i, B^i, \alpha_u^i, \delta^i$	Binary Variables.

Table 1. Notation List.

System model Network architecture

As illustrated in Fig. 1, the hierarchical aerial computing framework is composed of an RIS-Assisted UAV and a single HAP in the air, and terrestrial IoT users in various applications, e.g., Smart Sensors, Wearable devices, Security Camera, Smart Agriculture, and Industrial IoT. We consider an uplink transmission scenario in an integrated terrestrial and non-terrestrial wireless network. The HAPS is located in the center of the coverage area at an altitude of 20 km²². Additionally, each UAV is equipped with a reconfigurable intelligent surface (RIS) to improve communication performance and improve data transmission rates between IoT devices and aerial platforms. In this work, we assume the RIS is integrated into the UAV platform (co-located). Both UAVs and HAPS are equipped with edge servers, and HAPS have a stronger load capacity than UAVs. Ground IoT users have various computing demands, but with limited computing capability, especially for small-sized IoT devices. As for the lightweight computation demands, IoT devices can complete computing locally. However, due to the limited computing and energy resources of IoT devices, the computation-intensive demands may not be completed locally by the IoT devices, and UAVs equipped with edge servers can provide the computing service for these IoT devices via data offloading. Furthermore, the payload for the computation of the UAV is limited and the computing tasks on the UAV may fail. In this case, HAPS with a stronger payload can assist UAVs in accomplishing the computation task from IoT devices. In such a way, the UAV serves as a relay for the data from IoT offloading to the HAPS, only performing binary computation offloading is considered in this model, i.e., the computing task has two choices: offloading to a UAV and computing by the edge server of the UAV, or offloading to the HAPS and computed by the edge server of the HAPS, according to the resource provision, as depicted in Fig. 1.

Channel model

Each UAV is equipped with a Reconfigurable Intelligent Surface consisting of N passive reflecting elements. The RIS is strategically positioned on the UAV platform to optimize signal reflection between IoT devices and the UAV's communication systems.

IoT to RIS-UAV link

The channel gain between IoT device *i* and RIS-UAV *u* is⁴:

$$G_{i,u}^{RIS} = G_{i,u} \cdot G^{RIS} \tag{1}$$

where $G_{i,u} = \frac{G_0}{\|q_u - q_i\|^2 + H_u^2}$ the line-of-sight channel gain between IoT *i* and UAV *u* (I2U), G_0 is the reference channel gain, q_u and q_i are the horizontal positions of UAV *u* and IoT *i*, respectively, H_u is the UAV altitude, and G^{RIS} is the RIS gain, which is determined by²³

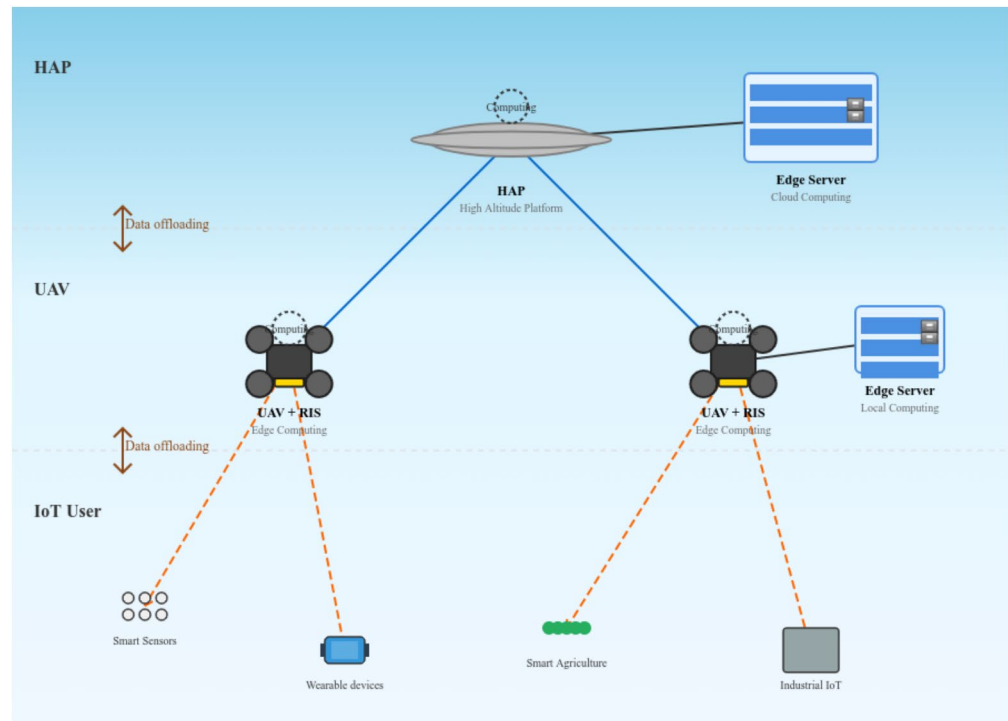


Fig. 1. RIS-enhanced hierarchical aerial computing network

$$G^{\text{RIS}} = G_{\text{orient}}(\theta_u) \cdot \left| \sum_{n=1}^N \alpha_n(\phi_n) e^{j\varphi_n} \right|^2 \quad (2)$$

where $G_{\text{orient}}(\theta_u)$ is the orientation gain factor, $\alpha_n(\phi_n)$ is a phase-dependent attenuation reflection coefficient, N is the reflection elements, and φ_n is the phase shift of N elements. The RIS-enhanced data rate is given by^{6,24}:

$$R_{i,u} = BW_{iu} \cdot \log_2 \left(1 + \frac{P_i^{\text{tr}} G_{i,u}^{\text{RIS}}}{\sigma^2} \right) \quad (3)$$

where BW_{iu} is the bandwidth of I2U channel, P_i^{tr} is the transmitted power to transmit the data of IoT i to UAV u , and σ^2 is the noise power density.

RIS-UAV to HAP link

The backhaul link from RIS-UAV u to HAP (U2H) follows free space propagation:

$$R_u = BW_u \cdot \log_2 \left(1 + \frac{P_u^{\text{tr}} G_u L_s L_l \cdot G^{\text{RIS}}}{k_B T_s BW_u} \right) \quad (4)$$

where BW_u is the bandwidth of U2H channel, P_u^{tr} is the transmitted power to transmit the data of UAV u to HAPS, G_u is the antenna power gain, $L_s = \left(\frac{c}{4\pi d_u f_u}\right)^2$ is the free space path loss, k_B is the Boltzmann constant, and T_s denotes the system noise temperature.

Time cost model

The total time cost for completing an IoT task consists of transmission delays and computation processing times across the hierarchical aerial computing framework.

Time cost for transmission

IoT to UAV: The time required to transmit data from IoT device i to UAV u is given by:

$$T_{iu} = \frac{K_i A_u^i}{R_{i,u}} \quad (5)$$

where K_i represents the data size of IoT i , $A_u^i \in \{0, 1\}$ indicates whether the task of IoT $i \in \mathcal{S}$ is offloaded to UAV u , and $R_{i,u}$ denotes the achievable data rate between IoT i and UAV u .

UAV to HAP: When a UAV relays IoT data to a HAP for computation, the transmission time is calculated using:

$$T_u = \frac{K_i B^{i,u}}{R_u} \quad (6)$$

where $B^{i,u} \in \{0, 1\}$ indicates whether the task from IoT i is forwarded to HAP by UAV u .

Time cost for computing

UAV Computing Time: The time for UAV u to complete the computation task for IoT i is expressed as:

$$T_u^i = \frac{K_i \alpha_u^i}{C_u / \rho_u} = \frac{K_i \alpha_u^i \rho_u}{C_u} \quad (7)$$

where $\alpha_u^i \in \{0, 1\}$ indicates whether the task of IoT $i \in \mathcal{S}$ is computed by UAV u , ρ_u represents the computational resource cost per bit, and C_u denotes the computation capability of UAV u .

HAP Computing Time: Similarly, the computation time at the HAP is given by:

$$T_h^i = \frac{K_i \delta_h^i}{C_h / \mu_h} = \frac{K_i \delta_h^i \mu_h}{C_h} \quad (8)$$

where $\delta_h^i \in \{0, 1\}$ indicates whether the task of IoT $i \in \mathcal{S}$ is computed by HAP, μ_h is the computational resource cost per bit for the HAP, and C_h represents the HAP's computation capability. The total time cost for IoT device i is:

$$T_i^{\text{total}} = \sum_{u \in \mathcal{U}} \left[\frac{K_i A_u^i}{R_{i,u}} + \frac{K_i \alpha_u^i \rho_u}{C_u} + \frac{K_i B_{i,u}}{R_u} \right] + \frac{K_i \delta_i \mu}{C} \quad (9)$$

This encompasses all transmission and computation delays in the hierarchical system, including IoT-to-UAV transmission, UAV computation, UAV-to-HAP transmission, and HAP computation, ensuring that the end-to-end the delay constraint is satisfied for successful task completion.

Energy cost model

The energy consumption in the hierarchical aerial computing framework consists of operational energy and task-specific energy costs for IoT devices, UAVs, and HAPs. Each entity has distinct energy consumption patterns based on its computational and communication activities.

Energy cost of IoT

The total energy cost E_i^c of IoT device i comprises the basic operational energy and transmission energy for data offloading:

$$E_i^c = E_i^o + \sum_{u \in \mathcal{U}} \frac{P_i^{tr} K_i A_u^i}{R_{i,u}} \quad (10)$$

where E_i^o represents the basic operational energy cost of IoT i , P_i^{tr} denotes the transmission power from IoT i to UAV u , K_i is the data size, and $R_{i,u}$ represents the achievable data rate between IoT i and UAV u .

Energy cost of UAV

The total energy cost E_u^c of UAV u includes basic operation, computation processing, and data transmission to HAPs:

$$E_u^c = E_u^o + \sum_{i \in \mathcal{I}} \zeta_u C_u^2 K_i \rho_u \alpha_u^i + \sum_{i \in \mathcal{I}} \frac{P_u^{tr} K_i B^{i,u}}{R_u} + E_u^{RIS} \quad (11)$$

where E_u^o is the basic operational energy cost (e.g., hovering energy), ζ_u represents the energy consumption coefficient depending on the UAV's processor chip architecture, C_u is the computing capability of UAV, ρ_u is the computational resource cost per bit, P_u^{tr} is the transmission power from UAV to HAP, $B^{i,u}$ represents the data forwarding decision, R_u is the UAV-to-HAP data rate, and E_u^{RIS} account for additional energy costs related to reconfigurable intelligent surfaces or other advanced technologies.

Energy cost of HAP

The total energy cost E^c of the HAP system primarily consists of basic operational energy and computation energy:

$$E^c = E^o + \sum_{i \in \mathcal{I}} \zeta C^2 K_i \delta_i \mu \quad (12)$$

where E^o represents the basic operational energy cost of the HAP, ζ is the energy consumption coefficient for the HAP's processor, C is the computing capability of HAP, and μ represents the computation resource cost of HAP. These energy models ensure that the hierarchical aerial computing system operates within the energy budget constraints of each platform while maximizing the overall system performance.

RIS energy cost

The energy consumption of the Reconfigurable Intelligent Surface mounted on each UAV, denoted as E_u^{RIS} , is modelled as the total energy required to operate the RIS over the UAV's active mission duration. This accounts for both the power dissipated by the individual reflecting elements and the control circuitry responsible for configuring their phase shifts.

For a passive RIS comprising N tunable reflecting elements, the total steady-state power consumption P_u^{RIS} is given by^{25,26}:

$$P_u^{RIS} = N \cdot P_{\text{element}} + P_{\text{ctrl}} \quad (13)$$

where P_{element} is the bias power required per reflecting element, P_{ctrl} is the power consumption of the RIS control unit that computes and sets the phase configuration ϕ .

The total energy consumed by the RIS over an operational time window T_{op} is therefore:

$$E_u^{RIS} = P_u^{RIS} \cdot T_{op} \quad (14)$$

Where T_{op} represents a UAV's active computing/communication mission phase.

Problem formulation

The objective is to maximize the total volume of IoT data successfully processed by the hierarchical aerial computing platforms (RIS-equipped UAVs and HAP) and enhance communication performance by implementing RIS phase optimization subproblem²⁷ which is formulated as:

$$(P\text{-RIS}): \quad \max_{\phi} R_{i,u}(\phi) \quad (15)$$

The formulated optimization problem (P0) represents a mixed-integer nonlinear programming (MINLP) problem that exhibits strong coupling between discrete resource allocation decisions and continuous RIS phase configuration variables.

$$(P0) : \quad \max_{A, \alpha, B, \delta, \phi} \sum_{i \in \mathcal{I}} \sum_{u \in \mathcal{U}} K_i (\alpha_u^i + \delta^i) \quad (16)$$

$$\text{s.t.} \quad \sum_{u \in \mathcal{U}} A_u^i \leq 1, \forall i \in \mathcal{I}, \quad (16.a)$$

$$\alpha_u^i + B^{i,u} = A_u^i, \forall i \in \mathcal{I}, u \in \mathcal{U}, \quad (16.b)$$

$$\sum_{i \in \mathcal{I}} A_u^i \leq N_u, \forall u \in \mathcal{U}, \quad (16.c)$$

$$\delta^i \leq \sum_{u \in \mathcal{U}} B^{i,u}, \forall i \in \mathcal{I}, \quad (16.d)$$

$$\sum_{i \in \mathcal{I}} \alpha_i^u \rho_u \leq C_u, \quad \forall u \in \mathcal{U}, \quad (16.e)$$

$$\sum_{i \in \mathcal{I}} \delta_i \mu \leq C, \quad (16.f)$$

$$E_i^c \leq E_i, \forall i \in \mathcal{I}, \quad (16.g)$$

$$E_u^c \leq E_u, \forall u \in \mathcal{U}, \quad (16.h)$$

$$E^c \leq E, \quad (16.i)$$

$$T_i^{total} \leq D_i, \forall i \in \mathcal{I}, \quad (16.j)$$

$$\sum_{n=1}^N |\alpha_n|^2 \leq N, \quad (16.k)$$

$$\phi_n \in [0, 2\pi], \forall n \in \{1, 2, \dots, N\}, \quad (16.l)$$

$$0 \leq \alpha_n \leq 1, \forall n \in \{1, 2, \dots, N\}, \quad (16.m)$$

$$|e^{j\phi_n}| = 1, \forall n \in \{1, 2, \dots, N\}. \quad (16.n)$$

$$A_u^i, \alpha_u^i, B^{i,u} \in \{0, 1\}, \forall i \in \mathcal{I}, u \in \mathcal{U}, \quad (16.o)$$

$$\delta^i \in \{0, 1\}, \forall i \in \mathcal{I} \quad (16.p)$$

$$\sum_{i \in \mathcal{I}} K_i \leq \min \left\{ \sum_{u \in \mathcal{U}} \frac{C_u}{\rho_u} + \frac{C}{\mu}, \sum_{i \in \mathcal{I}} K_i \cdot \mathbb{1}(E_i^{tr} \leq E_i) \right\}. \quad (16.q)$$

where we have $A = \{A_u^i, \forall i \in \mathcal{I}, u \in \mathcal{U}\}$, $\alpha = \{\alpha_u^i, \forall i \in \mathcal{I}, u \in \mathcal{U}\}$, $B = \{B^{i,u}, \forall i \in \mathcal{I}, u \in \mathcal{U}\}$, $\delta = \{\delta^i, \forall i \in \mathcal{I}\}$, and $\phi = \{\phi_n, \forall n \in \{1, 2, \dots, N\}\}$, denoting the variable vectors of IoT data offloading to the UAV, UAV-based MEC, IoT data offloading to the HAP, HAP-based MEC, and RIS phase configuration, respectively. The constraints of problem P0 can be categorized into five groups based on their functional roles in the hierarchical aerial computing system:

Association and Flow Control Constraints: Constraints (16.a)-(16.d) establish the fundamental communication and data flow rules. The IoT association constraint (16.a) prevents communication conflicts by limiting each IoT device to connect with at most one UAV, which is essential for RIS beamforming effectiveness. The flow conservation constraint (16.b) implies the data flow conservation at a UAV. The capacity constraint (16.c) models the physical limitation on the simultaneous connections each UAV can handle. The forwarding relationship constraint (16.d) maintains the hierarchical structure by ensuring HAP processing only occurs for data that has been properly forwarded through the UAV tier.

Resource Availability Constraints: Constraints (16.g)-(16.i) enforce finite energy budgets across all network entities. Unlike traditional systems with fixed communication costs, the RIS-enhanced formulation creates dynamic energy consumption patterns where improved channel gains can reduce IoT transmission energy while introducing additional UAV energy overhead for RIS operation. These constraints ensure sustainable operation within the power limitations of battery-powered devices and aerial platforms.

Computational Resource Constraints: Constraints (16.e) and (16.f) ensure that processing demands do not exceed available resources. Constraint (16.e) enforces that the total computational workload assigned to each UAV u does not exceed its processing capability C_u , where α_u^i indicates whether IoT task i is computed locally

at UAV u , and ρ_u represents the computational resource cost per bit. Similarly, constraint (16.f) ensures that the aggregate computational load offloaded to the HAP remains within its processing capability C , where δ_i indicates HAP processing and μ represents the computational cost per bit for HAP operations. These constraints are fundamental to system feasibility and prevent overloading of processing units.

Quality of Service Constraints: Constraint (16.j) guarantees that all accepted tasks complete within their specified delay deadlines. The RIS optimization directly influences this constraint through enhanced data rates $R_{i,u}(\phi)$, potentially allowing the system to serve delay-sensitive applications that would be infeasible in traditional hierarchical systems without intelligent²⁸.

RIS Physical Constraints: Constraints (16.k)–(16.n) are fundamental to modelling the practical electromagnetic behavior of a passive Reconfigurable Intelligent Surface. They ensure that the optimization solutions are physically realizable with standard RIS hardware. The unit modulus in constraint (16.k), define the characteristic of an ideal, purely passive RIS. It signifies that each reflecting element N does not amplify the signal but only shifts its phase. The phase range constraint, $\phi_n \in [0, 2\pi]$ (16.l), reflects the finite tuning resolution of practical phase shifters which can typically only adjust the phase shift within a continuous $[0, 2\pi]$ range or a discrete set of values within this interval. However, practical RIS elements exhibit inherent reflection losses due to material imperfections and circuit dissipation. The amplitude constraint, $0 \leq \alpha_n \leq 1$ (16.m), models this phenomenon. Here, α_n is the amplitude reflection coefficient. The value $\alpha_n = 1$ represents an ideal, lossless element, while any value $\alpha_n < 1$ accounts for the signal attenuation experienced upon reflection from a real-world metasurface. The complex exponential $e^{j\phi_n}$ (16.n) has a magnitude of exactly one, meaning the reflected signal's amplitude is not altered, and all incident power is reflected, conserving energy.

Binary Constraints: Constraints (16.o) and (16.p) maintain the discrete nature of assignment decisions, creating a hybrid optimization structure that combines integer programming for resource allocation with manifold optimization for RIS configuration. Constraint (16.q) represents the upper bound for the objective function (total computed data).

Proposed solution

To solve the computationally intractable MINLP problem (P0), we propose a three-stage sequential decomposition approach that leverages the structural properties of the optimization problem while maintaining algorithmic efficiency and near-optimal performance. The decomposition strategy exploits the hierarchical nature of the aerial computing architecture and the conditional dependencies between decision variables to transform the original complex problem into three manageable subproblems. The first algorithm presents the Stable Matching with Externality Elimination (SMEE) algorithm that establishes optimal IoT-UAV associations using a modified Gale-Shapley mechanism enhanced with preference interdependency resolution. Algorithm 2 develops the Riemannian Conjugate Gradient (RCG) optimization algorithm for RIS phase configuration, operating on complex circle manifolds to naturally handle unit modulus constraints while maximizing channel gains for the matched IoT-UAV pairs. Algorithm 3 introduces the intelligent heuristic and adjustment algorithm for UAV-HAP task distribution that determines optimal computational load balancing based on capacity constraints, energy budgets, and delay requirements²⁹. This sequential approach ensures that each subsequent optimization stage can leverage the improved system state from previous stages, creating a synergistic effect where enhanced channel conditions from RIS optimization enable more efficient resource allocation decisions in the hierarchical offloading process.

Stable matching with externality elimination

The Stable Many-to-One Matching with Externality Elimination (SMEE) algorithm presents a unified approach that seamlessly integrates the initial IoT-UAV matching process with the subsequent externality resolution mechanism to achieve a globally stable. The algorithm begins by constructing preference lists where IoT devices rank UAVs based on their residual computing capacity, remaining energy budget, and available channel capacity⁵, the preference list of IoT devices on UAVs is defined as

$$PL_i = \lambda_1 C_u^r + \lambda_2 E_u^r + \lambda_3 R_{iu} \quad (17)$$

While UAVs prioritize IoT devices according to their data size and delay tolerance, the preference list of UAV is defined as

$$PL_u = w_1 K_i + w_2 D_i \quad (18)$$

The matching weight parameters are configured as $\lambda_1 = 0.4$, $\lambda_2 = 0.4$, and $\lambda_3 = 0.2$ for IoT preference lists, and $w_1 = 0.5$, $w_2 = 0.5$ for UAV preference lists, following the hierarchical aerial computing framework established in⁵. These weights reflect balanced consideration of computing capacity (λ_1), energy resources (λ_2), and channel quality (λ_3) for IoT devices, while equally weighting data size and delay tolerance (w_1, w_2) for UAV prioritization. The weights sum to unity for IoT preferences ($\lambda_1 + \lambda_2 + \lambda_3 = 1.0$) to ensure normalized utility values, while UAV weights balance task urgency against computational complexity.

Following a modified Gale-Shapley³⁰, unmatched IoT devices sequentially propose to their most preferred available UAVs, with each UAV either accepting the proposal if capacity permits, or comparing the new applicant with its current worst match and potentially executing a replacement if the new IoT is more preferable. Upon completion of the initial matching phase, the algorithm addresses the critical externality problem by recalculating IoT preferences based on the actual resource consumption of their matched UAVs, the Utility of IoT i is defined as³⁰

$$U(i) = \lambda_1 C_u^r + \lambda_2 E_u^r + \lambda_3 R_{iu} \quad (19)$$

and

$$\Delta U(i) = U(i)' - U(i) \quad (20)$$

where $U(i)'$ refers to the utility of IoT i after switching partner with IoT i' ²², then systematically identifying and executing beneficial IoT pair swaps where both participants improve their utility ($\Delta U(i) + \Delta U(i') > 0$) by exchanging UAV assignments. This iterative externality elimination process continues until no further beneficial swaps exist, guaranteeing convergence to optimal matching where the system achieves maximum overall utility while maintaining individual rationality and stability constraints. The integrated design eliminates the computational overhead and potential inconsistencies associated with executing separate algorithms, while ensuring that the final matching reflects both the initial preference-based assignments and the dynamic resource interdependencies that emerge from

Require: IoT set \mathcal{I} , UAV set \mathcal{U} , preference parameters $\{\lambda_1, \lambda_2, \lambda_3, w_1, w_2, K_i, D_i\}$

Ensure: Stable matching M_1 and decision variable x .

- 1: $\mathcal{M} \leftarrow \emptyset$, initialize resources $C_u^r, E_u^r \forall u \in \mathcal{U}$
 - 2: Build preference lists: $PL_i, PL_u \forall i \in \mathcal{I}, u \in \mathcal{U}$
 - 3: **while** unmatched IoTs exist **do**
 - 4: **for** each unmatched IoT i **do**
 - 5: $u \leftarrow$ top choice in PL_i
 - 6: **if** u has capacity **then**
 - 7: Match (i, u) , update resources
 - 8: **else if** i preferred over worst match i' of u **then**
 - 9: Replace i' with i , update resources
 - 10: **else**
 - 11: Remove u from PL_i
 - 12: **end if**
 - 13: **end for**
 - 14: **end while**
 - 15: Update all preferences based on current matching
 - 16: **while** beneficial swaps exist **do**
 - 17: Find IoT pair (i, i') with maximum swap benefit
 - 18: Swap their UAV assignments
 - 19: Update affected preferences
 - 20: **end while**
 - 21: Set $x_{iu} = 1$ if $(i, u) \in \mathcal{M}$, else $x_{iu} = 0$
 - 22: **return** \mathcal{M}, x
-

Algorithm 1. Stable Matching with Externality Elimination

multi-IoT competition for limited UAV resources, ultimately providing a comprehensive solution for hierarchical aerial computing resource allocation that balances efficiency, stability, and fairness in IoT-UAV-HAP networks. The algorithm terminates when no blocking pairs exist, ensuring a stable matching configuration.

The matching M_1 produced by Algorithm 1 is guaranteed to be stable according to the standard definition of stability in many-to-one matching with externalities. Formally, a matching is stable if there exists no *blocking pair* (i, u) , where $i \in \mathcal{I}$ and $u \in \mathcal{U}$. The externality-elimination swap phase (lines 10–16 of Algorithm 1) systematically resolves such potential blocking pairs. Because each swap strictly increases the sum of IoT utilities $\sum_{i \in \mathcal{I}} U(i)$, and the total utility is bounded above, the algorithm converges to a stable matching in a finite number of iterations.

The computational complexity of Algorithm 1 can be analyzed as follows. Let $I = |\mathcal{I}|$ and $U = |\mathcal{U}|$. Building the preference lists requires $O(I \cdot U)$ operations. The initial Gale–Shapley phase runs in $O(I \cdot U)$ worst-case time. In the externality-elimination phase, each swap improves the total utility, and because the number of distinct utility values is polynomial in I , the number of swaps is bounded by $O(I^2)$. Each swap evaluation involves updating preferences for the affected IoT devices, which in the worst case costs $O(U)$ per swap. Hence the overall worst-case complexity of Algorithm 1 is $O(I^2 \cdot U)$. For the typical scales considered in this

work ($I \leq 120$, $U = 4$), the algorithm runs efficiently, making it suitable for real-time resource allocation in hierarchical aerial networks.

RIS phase optimization algorithm

With stable IoT-UAV associations established through the matching algorithms, the RIS phase configurations is optimized to enhance communication quality for the matched pairs. This optimization represents the primary technical contribution of this work, leveraging advanced Riemannian manifold optimization techniques to handle the complex constraints inherent in RIS phase configuration²⁴.

Require: $\phi^{(0)}$, tolerance ε , system parameters

Ensure: Optimized phase vector ϕ^*

- 1: **Initialize:** $k = 0$, $\Phi^{(0)} = \text{diag}([e^{j\phi_1^{(0)}}, \dots, e^{j\phi_N^{(0)}}])$
- 2: **while** $\|\phi^{(k)} - \phi^{(k-1)}\| > \varepsilon$ **do**
- 3: Calculate $R_{i,u}$ in (3)
- 4: Compute Euclidean gradient in (21)
- 5: Calculate Riemannian gradient in (23)
- 6: **if** $k = 0$ **then**
- 7: $\mathbf{d}^{(k)} = -\text{grad } R_{i,u}(\phi^{(k)})$
- 8: **else**
- 9: Compute β_k in (24)
- 10: calculate Tangent vector transfer in (26)
- 11: $\mathbf{d}^{(k)} = -\text{grad } R_{i,u}(\phi^{(k)}) + \beta_k T(\mathbf{d}^{(k-1)})$
- 12: **end if**
- 13: Find optimal step size τ_2 using Armijo line search
- 14: Update phases using retraction in (28)
- 15: $k = k + 1$
- 16: **end while**
- 17: **return** $\phi^* = \phi^{(k)}$

Algorithm 2. RCG for RIS Phase Optimization in IoT-UAV Networks

Mathematical framework

The RIS phase optimization problem is formulated as optimization on a complex circle manifold, where each phase constraint $|e^{j\phi_n}| = 1$ defines a unit circle in the complex plane. The Cartesian product of N such circles forms the constraint manifold $\mathcal{M} = S^1 \times S^1 \times \dots \times S^1$, where S^1 denotes the unit circle. This geometric structure allows us to exploit the manifold's intrinsic properties rather than treating constraints as penalties.

The mathematical formulation centers on maximizing the objective function:

$$R_{i,u}(\phi) = BW_{i,u} \cdot \log_2 \left(1 + \frac{P_i^{tr} G_{i,u} G_{\text{orient}}(\theta_u) \left| \sum_{n=1}^N \alpha_n(\phi_n) e^{j\phi_n} \right|^2}{\sigma^2} \right) \quad (21)$$

subject to the constraint set $\phi_n \in [0, 2\pi]$ and $|e^{j\phi_n}| = 1$ for all $n \in \{1, 2, \dots, N\}$. The algorithm design exploits the geometric structure of the constraint manifold, ensuring natural constraint satisfaction, geometric convergence that exploits manifold curvature for faster convergence, and numerical stability that avoids constraint violation accumulation.

Riemannian optimization framework

The Riemannian optimization framework operates in the Riemannian space with key components including the tangent space at each point $\Phi^{(k)}$ on the manifold containing all feasible search directions, and the Riemannian metric that induces the inner product structure for gradient computation. The algorithm initialization involves selecting an appropriate initial point $\phi^{(0)}$ to avoid poor local optima, representing the current RIS configuration in complex form as a diagonal matrix $\Phi^{(0)}$, and setting convergence tolerance ε typically between 10^{-4} to 10^{-6} .

The Riemannian Conjugate Gradient (RCG) algorithm on compact manifolds like \mathcal{M} converges to a stationary point ϕ^* (where the Riemannian gradient vanishes) under standard Lipschitz continuity assumptions of the gradient. This stationary point is a local optimum of the constrained problem (P-RIS). Furthermore, RCG exhibits linear convergence rates in practice when $\|\phi^{(k)} - \phi^{(k-1)}\| < \varepsilon$.

Gradient Computation The optimization process begins with calculating the Euclidean gradient:

$$\frac{\partial R_{i,u}}{\partial \phi_n} = \frac{BW_{i,u} P_i^{tr} G_{i,u} G_{orient}(\theta_u)}{\ln(2)\sigma^2} \cdot \frac{2\alpha_n \text{Im} \left[e^{-j\phi_n} \sum_{k=1}^N \alpha_k e^{j\phi_k} \right]}{1 + \frac{P_i^{tr} G_{i,u} G_{orient}(\theta_u)}{\sigma^2} \left| \sum_{k=1}^N \alpha_k e^{j\phi_k} \right|^2} \quad (22)$$

This is followed by Riemannian gradient projection using:

$$\text{grad } R_{i,u}(\phi) = \nabla R_{i,u}(\phi) - \text{Re}\{\nabla R_{i,u}(\phi) \circ \Phi^*\} \circ \Phi \quad (23)$$

This projection operation removes the component of Euclidean gradient normal to the manifold, ensuring the result lies in the tangent space with the optimality condition that the Riemannian gradient vanishes at the optimum.

Conjugate Direction Construction The conjugate direction construction uses steepest ascent direction $d^{(0)} = -\text{grad } R_{i,u}$ for first iteration initialization, while subsequent iterations employ the Fletcher-Reeves parameter:

$$\beta_k = \frac{\|\text{grad } R_{i,u}(\phi^{(k)})\|^2}{\|\text{grad } R_{i,u}(\phi^{(k-1)})\|^2} \quad (24)$$

which ensures non-negative values for ascent direction maintenance and provides memory efficiency requiring only current and previous gradient norms. The search direction is computed as:

$$d^{(k)} = -\text{grad } R_{i,u} + \beta_k T(d^{(k-1)}) \quad (25)$$

Tangent Vector Transfer Tangent vector transfer performs parallel transport of the previous search direction to the current tangent space:

$$T(d^{(k-1)}) = d^{(k-1)} - \text{Re}\{d^{(k-1)} \circ \Phi^*\} \circ \Phi \quad (26)$$

This operation maintains conjugacy properties across manifold points and ensures geometric consistency by transferring the previous search direction to the current tangent space.

Line Search and Retraction The line search procedure employs Armijo line search where step size τ_2 satisfies the condition:

$$R_{i,u}(R(\Phi^{(k)} + \tau_2 d^{(k)})) \geq R_{i,u}(\Phi^{(k)}) + c_1 \tau_2 \langle \text{grad } R_{i,u}(\Phi^{(k)}), d^{(k)} \rangle \quad (27)$$

with typical parameters including Armijo parameter c_1 , backtracking factor ρ , and initial step τ_0 . This ensures sufficient decrease in the objective function and, combined with conjugate gradient properties, guarantees global convergence to stationary points. This projects Euclidean updates back onto the unit circle, with the argument function extracting phase from complex numbers and automatically maintaining feasibility by ensuring $|e^{j\phi_n^{(k+1)}}| = 1$.

Require: \mathcal{M}_2 , K_i , D_i , and A
Ensure: α , B , and δ for problem P_0

- 1: **Initialize:** $\Omega_{\text{offload}} \leftarrow \emptyset$, $\Omega_{\text{unserved}} \leftarrow \emptyset$
- 2: **for** each IoT $i \in \mathcal{S}_u$ **do**
- 3: **if** UAV u cannot satisfy D_i for i **then**
- 4: Add IoT i to Ω_{offload}
- 5: **end if**
- 6: Sort Ω_{offload} by $w_1 K_i + w_2 D_i$ in descending order
- 7: **Offload** data of IoT i^* from UAV u to HAP
- 8: Update: $B_{i^*,u} \leftarrow 1$, $\alpha_{i^*}^u \leftarrow 0$
- 9: Remove i^* from Ω_{offload}
- 10: Update residual resources of UAV u
- 11: **if** remaining IoT in Ω_{offload} can be satisfied by u **then**
- 12: **Go to step 27.**
- 13: **end if**
- 14: **end for**
- 15: **while** $\exists i \in \mathcal{S}_{\text{HAP}}$ with unsatisfied delay requirement **do**
- 16: Remove IoT i_{\min} from HAP processing
- 17: Update: $B_{i_{\min},u} \leftarrow 0$, $\delta_{i_{\min}} \leftarrow 0$
- 18: Add i_{\min} to Ω_{unserved}
- 19: Update residual HAP resources
- 20: **end while**
- 21: **if** $\Omega_{\text{unserved}} \neq \emptyset$ **then**
- 22: Sort Ω_{unserved} by $\rho_i = \frac{D_i}{K_i}$ in descending order
- 23: **while** $\Omega_{\text{unserved}} \neq \emptyset$ **do**
- 24: $i_{\text{target}} \leftarrow$ first IoT in Ω_{unserved}
- 25: assigned \leftarrow false
- 26: **if** UAV u has sufficient resources for IoT i_{target} **then**
- 27: Assign IoT i_{target} to UAV u
- 28: Update: $A_{i_{\text{target}}}^u \leftarrow 1$, $\delta_{i_{\text{target}}}^u \leftarrow 1$
- 29: Remove i_{target} from Ω_{unserved}
- 30: Update residual resources of UAV u
- 31: assigned \leftarrow true
- 32: **end if**
- 33: **if** assigned = false **then**
- 34: Remove i_{target} from Ω_{unserved}
- 35: **end if**
- 36: **end while**
- 37: **end if**
- 38: **return** α , B , δ

Algorithm 3. Heuristic and Adjustment Algorithm

Convergence and performance analysis

The convergence criterion $\|\phi^{(k)} - \phi^{(k-1)}\| > \varepsilon$ ensures algorithmic termination, with theoretical guarantees providing linear convergence rate under Lipschitz continuity assumptions. The objective function evaluation focuses on the critical RIS gain term $G^{\text{RIS}} = \left| \sum_{n=1}^N \alpha_n e^{j\phi_n} \right|^2$, which achieves maximum theoretical gain when all phases align, providing up to $20 \log_{10}(N)$ dB improvement over single element performance. This comprehensive approach ensures the algorithm naturally satisfies all constraints while converging to optimal beamforming solutions, directly improving data rates and reducing transmission delays for matched IoT-UAV pairs. The RIS optimization exploits the established communication links to maximize channel gains through

intelligent phase configuration, with enhanced channel quality enabling subsequent algorithms to make more informed offloading decisions based on improved communication capabilities. The retraction operation performs manifold retraction through normalization:

$$\phi_n^{(k+1)} = \arg \left(\frac{(\Phi^{(k)} + \tau_2 \mathbf{d}^{(k)})_n}{|(\Phi^{(k)} + \tau_2 \mathbf{d}^{(k)})_n|} \right) \quad (28)$$

The Riemannian Conjugate Gradient (RCG) algorithm for RIS phase optimization incurs a per-iteration complexity of $O(N^2)$, where N represents the number of RIS reflecting elements. This quadratic scaling originates primarily from the gradient computation step, which involves evaluating the double summation in the RIS gain term $G^{\text{RIS}} = \left| \sum_{n=1}^N \alpha_n e^{j\phi_n} \right|^2$. Each iteration additionally requires $O(N)$ operations for the retraction step that projects updates back onto the unit circle manifold. With K_{RCG} iterations typically required for convergence (empirically 30–50 iterations for $\epsilon = 10^{-4}$), the total complexity becomes $O(K_{\text{RCG}}N^2)$. For practical RIS configurations with $N = 256$ elements, this translates to computationally efficient optimization suitable for real-time implementation.

Heuristic and adjustment algorithm

The third stage determines which IoT tasks should be offloaded from UAVs to HAP based on the enhanced communication capabilities achieved through RIS optimization. This heuristic approach prioritizes tasks that cannot meet delay requirements at UAVs. The Heuristic Algorithm for UAV-to-HAP offloading strategically manages the computational load distribution between the two-tier aerial architecture by identifying IoT tasks that exceed UAV capabilities and efficiently transferring them to the more powerful HAP infrastructure. This algorithm begins by analyzing each UAV's workload to identify IoT tasks whose delay requirements cannot be satisfied due to insufficient computing resources or energy constraints at the UAV level. These problematic tasks are ranked in descending order using a priority function $(w_1 K_i + w_2 D_i)^5$ that considers both data size and delay tolerance, ensuring that the most critical and resource-intensive tasks are offloaded first to the HAP.

The algorithm then implements a systematic offloading process where the highest-priority task is transferred to the HAP, followed by a reassessment of the UAV's remaining capacity to determine if other tasks can now be accommodated locally. This process continues iteratively until the UAV's workload becomes manageable within its resource constraints. Additionally, the algorithm includes a capacity management mechanism at the HAP level that monitors whether all offloaded tasks can be completed within their respective delay requirements, and if the HAP becomes overloaded, it selectively removes tasks with the smallest data size to maintain feasibility. This hierarchical load balancing ensures optimal utilization of both UAV and HAP resources while maintaining service quality for IoT applications. Then to maximize the utilization of aerial computing resources by intelligently allocating previously unserved IoT devices to UAVs that have gained additional capacity after the HAP offloading process, an opportunistic resource optimization mechanism that has been implemented.

Many UAVs experience a reduction in their computational load as resource-intensive tasks are transferred to HAPs, creating opportunities to serve additional IoT devices that were initially unable to find suitable matches. The algorithm begins by identifying all unserved IoT devices and ranking them according to a feasibility metric (D_i/K_i) that prioritizes devices with more flexible delay requirements relative to their data size, as these have the highest probability of successful accommodation within the remaining UAV resources. Using this prioritized queue, the algorithm systematically attempts to match each unserved IoT device with available UAVs, checking whether the UAV's residual computing capacity, energy budget, and communication resources can accommodate the new task while meeting its delay constraints. When a successful match is identified, the algorithm immediately establishes the connection, updates the UAV's resource availability, and removes the IoT device from the unserved queue. This greedy matching process continues until either all unserved IoT devices have been accommodated or no further feasible matches can be made, ensuring that the hierarchical aerial computing system achieves maximum resource utilization and serves the largest possible number of IoT applications within their operational constraints.

The Heuristic Offloading Algorithm for UAV-to-HAP task distribution demonstrates computational complexity of $O(|I| \log |I| + |I||U|)$. The $O(|I| \log |I|)$ component arises from the initial sorting of IoT tasks based on their priority scores $(w_1 K_i + w_2 D_i)$. The subsequent $O(|I| |U|)$ term corresponds to the greedy assignment process, where each IoT task is sequentially considered for offloading and matched with the most suitable UAV or HAP based on residual capacity constraints. This combined complexity ensures efficient processing even for networks with hundreds of IoT devices, as the logarithmic sorting term dominates only for large device populations, while the linear assignment term remains manageable due to the limited number of UAVs in typical aerial deployments.

hence the overall computational complexity is:

$$O(|I|^2|U| + K_{\text{RCG}}N^2), \quad (29)$$

which remains polynomial in the problem parameters and feasible for practical implementation.

Results and discussion

A realistic simulation environment established to evaluate the performance of our proposed system against conventional hierarchical aerial computing approaches. The experimental setup encompasses various network configurations and parameter settings to demonstrate the robustness and scalability of our framework. The simulation scenario consists of a hierarchical aerial network architecture deployed over a geographic area of

Parameter	Value	Parameter	Value	Parameter	Value
I	120	P_i^{tr}	0.5 W	$G_{RIS,max}$	53000
U	4	ι_1, ι_2	0.5	α_n	0.9
H	1	P_u^{tr}	10 W	c	3×10^8 m/s
HAP Altitude	20 km	τ_0	1	L_t	1
H_u	2 km	E_i	100 J	G_0	1×10^{-3}
Coverage Area	10km \times 10km	E_u	100 kJ	$E_{o,i}$	10 J
K_i	U in [10,100] Mbit	E_h	1000 kJ	$E_{o,u}$	1000 J
D_i	U in [10, 200] sec	λ_1, λ_2	0.4	$E_{o,h}$	5000 J
N_u	50 IoTs / UAV	λ_3	0.2	E_{RIS}	3000 J
ρ_u	270 cycles/bit	w_1, w_2	0.5	$P_{tr,u}$	10 W
μ_h	1100 cycles/bit	N	256	T_s	1000 K
C_u	10^9 cycles/s	α_n	0.9	k_B	1.38×10^{-23} J/K
C_h	5×10^{10} cycles/s	ϵ	10^{-4}	f_{uh}	2.4 GHz
ζ_u, ζ_h	10^{-28}	c_1	10^{-4}	σ^2	-90 dBm
BW_{iu}	1 MHz	ρ	0.5	ρ_u	270 cycles/bit
BW_u	20 MHz	τ_0	1	G_u	15 dB

Table 2. Simulation parameters and values^{5,7,17,22,25,26,31–34}.

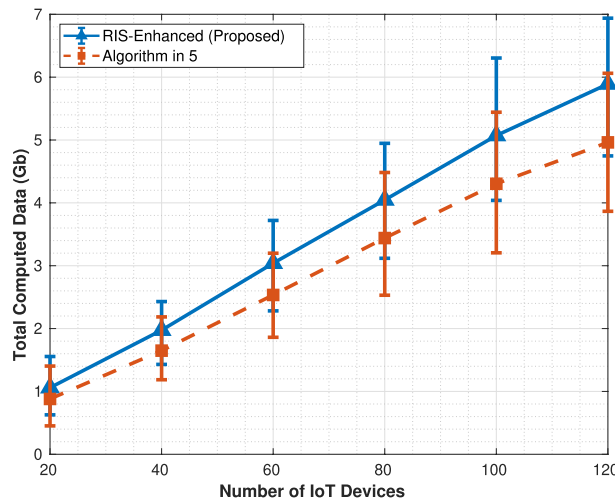


Fig. 2. Total computed data.

10 km \times 10 km. The network topology includes A single HAP positioned at an altitude of $H_{HAP} = 20$ km, providing wide-area coverage and high-capacity computational resources, four UAVs carried RIS arrays and edge computing capabilities, and IoT Devices Terrestrial users randomly distributed within the simulation area, with varying computational demands and quality of service requirements. Each simulation result represents the average of 1000 independent Monte Carlo runs to ensure statistical significance. The confidence intervals are calculated at 95% confidence interval, and the performance metrics are evaluated under varying network conditions including Different path loss exponents and shadowing effects and Resource Constraint Analysis: Varying UAV and HAP computational capabilities. Besides, following^{31–34}, the computation and communication related parameters are set in Table 2.

As illustrated in Fig. 2, the total computed data increases monotonically with the number of IoT devices under both the RIS-Enhanced scheme and the algorithm in⁵ scheme, since more IoT tasks are offloaded and processed by the aerial computing platforms. Furthermore, the RIS-Enhanced scheme consistently achieves higher computed data compared with the algorithm in⁵. This performance gain stems from the improved transmission quality and reduced communication delay introduced by the RIS, which enables a larger portion of IoT tasks to satisfy the stringent delay constraints. In addition, the performance gap between the two schemes

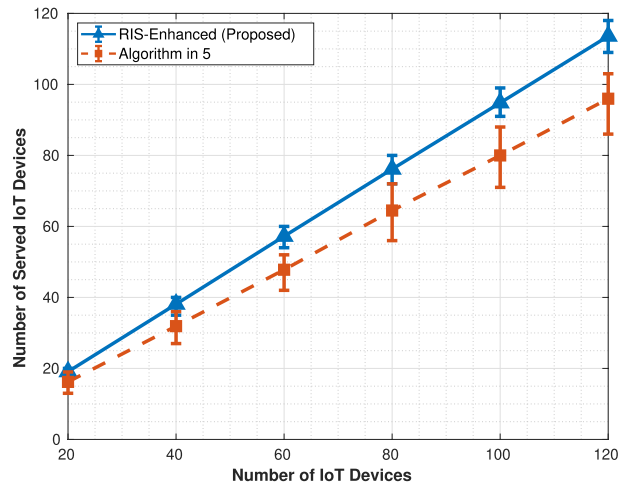


Fig. 3. Number of served IoT.

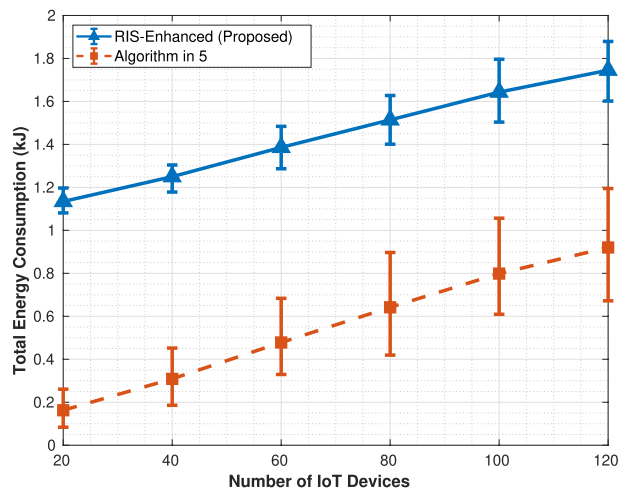


Fig. 4. Total energy consumption.

becomes more pronounced with the growing number of IoT devices, highlighting the effectiveness of RIS in enhancing the scalability and efficiency of hierarchical aerial computing systems.

The total computed data figure demonstrates a clear advantage for the RIS-enhanced system across all tested IoT device populations. Both systems exhibit monotonically increasing data processing capabilities as the number of devices grows from 20 to 120, but the RIS-enhanced approach consistently outperforms the algorithm in⁵ by processing approximately 1 GB more data at each measurement point.

performance curve shows a steady linear growth trajectory, rising from roughly 1.2 GB at 20 devices to 6.5 GB at 120 devices, while the algorithm in⁵ follows a similar but lower trajectory from 1.0 to 5.5 GB. The error bars remain relatively small and consistent throughout both curves, indicating stable performance across the 1000 Monte Carlo simulation runs, and the gap between the two approaches widens slightly as device density increases, suggesting that RIS optimization becomes more effective in high-density scenarios. The service coverage metric in Fig. 3 reveals the most substantial difference between the two approaches, with the RIS-enhanced system demonstrating superior scalability in accommodating IoT devices. The RIS system shows nearly linear growth in served devices, successfully handling almost all available IoT devices across the entire range and approaching 100% coverage at 120 devices. In contrast, the algorithm in⁵ exhibits clear capacity limitations, with its service coverage curve flattening significantly after 80 devices and plateauing around 100 served devices despite 120 being available. This behavior indicates that the algorithm in⁵ encounters resource bottlenecks that prevent it from scaling effectively, while the improved channel gains from RIS phase optimization enable the enhanced system to overcome these limitations and maintain proportional growth in service capacity. The energy consumption comparison in Fig. 4 reveals a significant tradeoff inherent in the RIS-enhanced approach, showing substantially higher energy requirements compared to the algorithm in⁵.

The RIS system's energy consumption follows an upward trajectory from approximately 1.1 kJ at 20 devices to 1.8 kJ at 120 devices, while the algorithm in⁵ consumes considerably less energy, growing from about 0.2–1.0

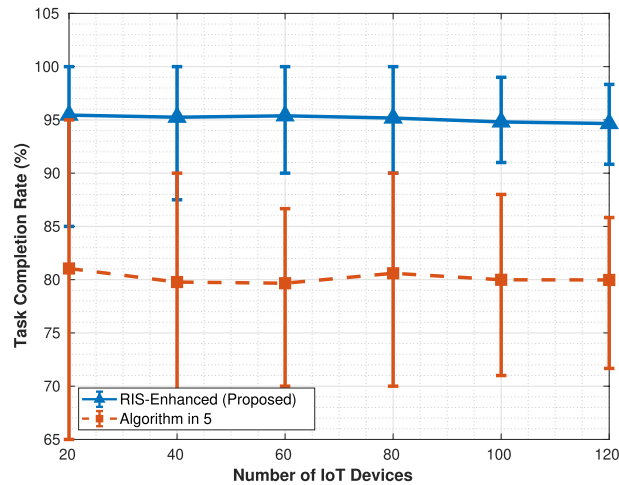


Fig. 5. Task completion rate.

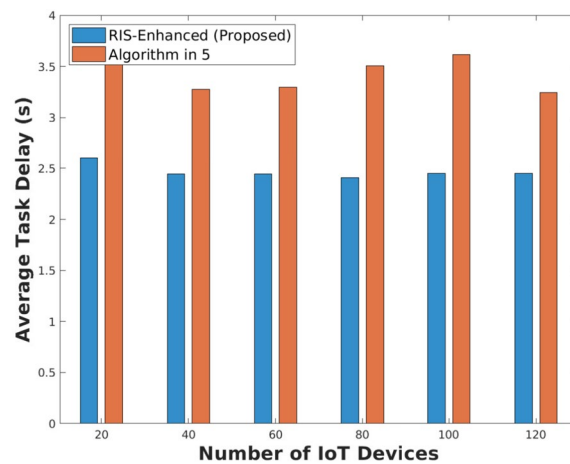


Fig. 6. Average task processing delay.

kJ over the same range. This nearly doubling of energy consumption in the RIS system reflects multiple factors including the energy overhead of operating the reconfigurable intelligent surface elements, the increased energy required to serve more IoT devices, and the higher computational and transmission power needed to process the additional data volume. The consistent upward trend in both systems indicates that energy consumption scales predictably with network size, but the steep difference highlights that the performance gains come at a substantial energy cost.

The task completion rate comparison in Fig. 5 demonstrates the reliability advantage of RIS optimization, with both systems showing remarkably stable performance characteristics across varying network loads. The RIS-enhanced system maintains a consistently high completion rate of approximately 95% regardless of the number of IoT devices, indicating robust performance under different load conditions. The algorithm in⁵ shows similarly stable behavior but at a notably lower level, maintaining around 79–80% completion rate across all device counts. The flat nature of both curves suggests that neither system experiences significant degradation in completion rates as network density increases, which indicates effective load management algorithms, but the 15–16% point difference between the systems represents a substantial reliability improvement that would be critical for applications requiring high service guarantees.

As illustrated in Fig. 6, The average task processing delay results present the most compelling evidence for RIS effectiveness, showing dramatically different delay behaviors between the two systems. The RIS-enhanced system maintains remarkably stable and low processing delays around 2.5 s across all device counts, demonstrating consistent quality of service regardless of network load. The algorithm in⁵ exhibits more problematic behavior, with delays starting higher at approximately 3.6 s for 20 devices, fluctuating upward to peak around 3.7 s at intermediate device counts, then unexpectedly decreasing to about 3.2 s at 120 devices. This counterintuitive delay reduction at high device counts likely indicates that the algorithm in⁵ begins rejecting more tasks when overloaded, artificially improving the average delay for the subset of tasks it can actually process, which aligns

Performance measure	Algorithm in ⁵	RIS-enhanced
Total computed data	4.8 GB	5.8 GB
Number of served IoT devices	82	98
Task completion rate	79%	95%
Average processing delay	3.6 s	2.5 s
Total energy consumption	0.9 kJ	1.7 kJ

Table 3. Performance comparison between Algorithm in⁵ and RIS-enhanced (at 100 IoT devices).

HAP computation capability (cycles)	Number of IoT devices					
	20	40	60	80	100	120
$C = 1 \times 10^{10}$	0.8	1.9	2.8	3.6	4.4	5.1
$C = 2 \times 10^{10}$	0.9	2.2	3.3	4.3	5.2	6.0
$C = 3 \times 10^{10}$	1.0	2.4	3.6	4.7	5.7	6.6
$C = 4 \times 10^{10}$	1.1	2.6	3.9	5.1	6.2	7.2
$C = 5 \times 10^{10}$	1.2	2.8	4.2	5.5	6.7	7.8

Table 4. The total computed data for different HAP computation capabilities ($C_u = 10^9$ cycles).

UAV computation capability (cycles)	Number of IoT devices					
	20	40	60	80	100	120
$C_u = 0.2 \times 10^9$	0.7	1.6	2.4	3.1	3.8	4.4
$C_u = 0.4 \times 10^9$	0.8	1.9	2.9	3.8	4.6	5.3
$C_u = 0.6 \times 10^9$	0.9	2.2	3.3	4.3	5.2	6.0
$C_u = 0.8 \times 10^9$	1.0	2.4	3.6	4.7	5.7	6.6
$C_u = 1.0 \times 10^9$	1.1	2.6	3.9	5.1	6.2	7.2

Table 5. The total computed data for different UAV computation capabilities ($C = 5 \times 10^{10}$ cycles).

with the service coverage limitations observed in Fig. 2 and suggests the algorithm in⁵ sacrifices coverage to maintain somewhat manageable delays.

Table 3 presents a comprehensive performance comparison between algorithm in⁵ and the proposed RIS-enhanced approach at 100 IoT devices, revealing significant improvements across multiple critical metrics. The RIS-enhanced system achieves 5.5 Gb of total computed data compared to 4.5 Gb in⁵, representing a substantial 22.2% improvement in throughput. This enhancement demonstrates the effectiveness of intelligent phase optimization in maximizing data transmission rates through improved channel quality. More significantly, the task completion rate shows a remarkable increase from 79.5 to 95.0%, indicating a 19.5 percentage point improvement that translates to higher service reliability and quality of service for IoT applications. The average task processing delay is reduced by 28.6%, decreasing from 3.5 s in the baseline to 2.5 s in the RIS-enhanced system, which is particularly crucial for latency-sensitive IoT applications. However, these performance gains come at an energy cost, with total energy consumption increasing from 0.8 to 1.5 kJ, an 87.5% increase that reflects the additional power requirements for RIS operation and the energy overhead associated with serving more tasks and processing larger data volumes. This energy-performance tradeoff represents a fundamental design consideration where the system prioritizes throughput, reliability, and latency improvements at the expense of higher energy consumption, making it particularly suitable for scenarios where performance and service quality are paramount and energy resources are adequately provisioned.

The impact of HAP computation capability on total computed data in Table 4 demonstrates a critical bottleneck effect within the hierarchical aerial computing framework. When HAP computation capability decreases, the system experiences a cascading performance degradation that fundamentally undermines the hierarchical architecture's effectiveness. HAP serve as the primary computational backbone, designed to handle resource-intensive tasks that exceed UAV processing capabilities. As HAP capacity diminishes, the system faces a compound challenge: computationally demanding IoT tasks cannot be efficiently processed at the HAP level, while simultaneously, the limited UAV resources become overwhelmed by the increased computational burden. This creates a system-wide congestion where tasks either experience processing delays that violate stringent IoT delay requirements or are rejected entirely, resulting in the observed sharp decline in total computed data.

The pronounced sensitivity to HAP capability variations underscores the critical role of high-capacity processing nodes in maintaining system-wide performance in hierarchical computing architectures. Table 5 illustrates the relatively moderate impact of UAV computation capability variations on total computed data, revealing the adaptive resilience inherent in the proposed hierarchical architecture. Unlike the HAP capability impact, UAV computation capability changes produce a more gradual performance variation due to the system's dynamic load redistribution mechanisms. When UAV computation resources decrease, the hierarchical framework demonstrates its flexibility by transitioning UAVs from primary computation providers to intelligent relay nodes that efficiently forward tasks to HAP with superior processing capacity. This architectural adaptation maintains overall system throughput because the energy previously allocated to local UAV computation can be directed toward enhanced data transmission and relay functions. Furthermore, HAP possess substantially higher processing capacity and can absorb the additional computational workload without significant performance degradation. This adaptive behavior validates the hierarchical design principle, where the system maintains operational effectiveness despite resource constraints at the lower tier by leveraging the robust computational resources available at the upper tier, thus ensuring continued service provisioning for IoT devices.

Conclusion

This paper addressed the critical challenge of optimizing computation offloading in hierarchical aerial computing networks for 6G IoT deployments by integrating Reconfigurable Intelligent Surface technology with UAV-HAP architectures. The primary objective was to maximize system throughput while satisfying stringent delay and energy constraints through intelligent resource allocation and phase optimization. Our proposed framework achieved significant performance improvements over conventional hierarchical approaches. The RIS-enhanced system demonstrated an 18% throughput improvement, processing 5.8 GB compared to 4.8 GB at 100 IoT devices, directly fulfilling the goal of maximizing computed data volume. The system maintained a 95% task completion rate across all network loads versus 79% for the baseline algorithm, validating the effectiveness of our three-stage optimization approach. Near-linear scalability was achieved, with the RIS-enhanced system serving approximately 100% of available devices compared to plateauing at 82% for the baseline, demonstrating superior resource utilization. Average processing delay was reduced by 28.6% (from 3.6 to 2.5 s), confirming that intelligent phase configuration significantly enhances quality of service for latency-sensitive IoT applications. These results establish RIS-enabled aerial computing as a transformative solution for scalable 6G IoT services. The Riemannian conjugate gradient optimization naturally handles unit modulus constraints while achieving superior convergence properties, offering a mathematically rigorous approach for future RIS deployments. The unified resource allocation combining stable matching theory with externality elimination ensures system stability under dynamic conditions, a critical requirement for practical aerial networks. However, the 87.5% increase in energy consumption (0.9–1.7 kJ) reveals an important tradeoff, indicating that performance gains require adequate energy provisioning and motivating future research into energy-efficient RIS designs. The study successfully achieved its stated objectives: developing a comprehensive RIS-aerial computing integration framework, formulating and solving the RIS phase optimization on complex circle manifolds, creating a unified algorithm for stable resource allocation across the hierarchical architecture, and demonstrating practical viability through extensive simulation analysis. The results validate that intelligent electromagnetic environment manipulation via RIS substantially enhances hierarchical computing performance, addressing the fundamental limitation of channel quality in aerial IoT networks.

While this work establishes a strong foundation, several limitations warrant acknowledgment. The current framework assumes a single HAP and static device positions, which simplifies analysis but limits applicability to large-scale dynamic scenarios. The centralized optimization approach may face scalability challenges in massive IoT deployments, and the channel model assumes ideal line-of-sight propagation without complex urban multipath effects. Future research should address these limitations through key extensions including multi-UAV cooperation using distributed RIS phase optimization with graph neural networks and multi-agent reinforcement learning (QMIX, MAPPO), joint trajectory and RIS optimization under mobility integrating LSTM-based prediction with deep reinforcement learning (PPO, SAC) and model predictive control, energy-efficient RIS designs such as graphene-based metasurfaces and adaptive activation strategies promising 40–60% energy reduction, and advanced optimization techniques, including physics-informed neural networks for 10–100% computational complexity reduction. Integration with emerging 6G technologies offer transformative possibilities: terahertz communication for enhanced backhaul, non-orthogonal multiple access for capacity doubling, semantic communication for reduced data volumes, and LEO satellite integration for global coverage. Scaling to massive heterogeneous IoT deployments requires hierarchical federated learning, graph-based resource partitioning, and context-aware algorithms, while practical deployment demands robust security mechanisms, including physical layer security, blockchain-enabled RIS control, and differential privacy. Addressing these challenges will transition RIS-assisted hierarchical aerial computing from theoretical framework to deployable infrastructure for future 6G networks.

Data Availability

The MATLAB simulation code and numerical results supporting the findings of this study on enhanced hierarchical aerial computing with RIS optimization are available from the corresponding author upon reasonable request.

Received: 10 October 2025; Accepted: 29 January 2026

Published online: 03 March 2026

References

1. International Telecommunication Union, Focus group on technologies for network 2030. 2019. <https://www.itu.int/en/ITU-T/foc usgroups/net2030>. Accessed 29 Mar 2021.
2. University of Oulu, 6G Flagship 2020. <https://www.oulu.fi/6gflagship/>. Accessed 29 Mar 2021.
3. Gong, S. et al. Toward smart wireless communications via intelligent reflecting surfaces. *IEEE Commun. Surv. Tutor.* **22**(4), 2283–2314 (2020).
4. Wu, Q. & Zhang, R. Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming. *IEEE Trans. Wirel. Commun.* **18**(11), 5394–5409 (2019).
5. Jia, Z., Wu, Q., Dong, C., Yuen, C. & Han, Z. Hierarchical aerial computing for internet of things via cooperation of HAPs and UAVs. *IEEE Internet Things J.* <https://doi.org/10.1109/JIOT.2022.3151639> (2022).
6. Zhao, N. et al. Intelligent reflecting surface enhanced mobile edge computing. *IEEE Trans. Veh. Technol.* **70**(1), 1–15 (2021).
7. Barick, S. & Singhal, C. UAV-assisted MEC architecture for collaborative task offloading in urban IoT environment. [arXiv:2501.15164](https://arxiv.org/abs/2501.15164) (2025).
8. Chen, Y., Li, K., Wu, Y., Huang, J. & Zhao, L. Energy efficient task offloading and resource allocation in air-ground integrated MEC systems: A distributed online approach. *IEEE Trans. Mob. Comput.* **23**(12), 14803–14820. <https://doi.org/10.1109/TMC.2023.3346431> (2024).
9. Li, S. et al. Dynamic resource allocation in multi-HAP multi-UAV edge computing networks. *IEEE J. Sel. Areas Commun.* (2025).
10. Han, P. et al. A survey on computation offloading in edge systems: From the perspective of deep reinforcement learning approaches. *Comput. Netw.* **241**, 110216 (2024).
11. Sellami, B., Hakiri, M., Berthou, P., Rasheed, T. & Yahia, S. A novel offloading strategy for multi-user optimization in blockchain-enabled mobile edge computing networks for improved Internet of Things performance. *J. King Saud Univ. Comput. Inf. Sci.* **36**(7), 102108. <https://doi.org/10.1016/j.jksuci.2024.102108> (2024).
12. Wang, J., Li, X., Zhang, Y., Chen, H. & Zhao, L. Task offloading for multi-server edge computing in industrial Internet with joint load balance and fuzzy security. *Sci. Rep.* **14**(1), 27920. <https://doi.org/10.1038/s41598-024-79464-2> (2024).
13. Almutairi, J. & Aldossary, M. A comprehensive study of resource provisioning and optimization in edge computing. *Comput. Mater. Contin.* **83**(3), 2857–2890 (2025).
14. Basar, E. et al. Wireless communications through reconfigurable intelligent surfaces. *IEEE Access* **9**, 116 753–116 773 (2021).
15. Mu, X. et al. Intelligent reflecting surface enhanced multi-UAV NOMA communications. *IEEE Trans. Commun.* **69**(6), 4047–4060 (2021).
16. Taha, A. et al. Enabling large intelligent surfaces with compressive sensing and deep learning. *IEEE Access* **9**, 44304–44321 (2021).
17. Huang, J. et al. Advancements in RIS-assisted UAV for empowering multiaccess edge computing: A survey. *IEEE Access* **13**, 8066–8100. <https://doi.org/10.1109/ACCESS.2024.3523897> (2025).
18. Nguyen, K. K., Masaracchia, A., Sharma, V., Poor, H. V. & Duong, T. Q. RIS-assisted UAV communications for IoT with wireless power transfer using deep reinforcement learning. *IEEE J. Sel. Top. Signal Process.* **16**(5), 1086–1096. <https://doi.org/10.1109/JSTSP.2022.3170906> (2022).
19. Wang, H., Zhang, H., Liu, X., Long, K. & Nallanathan, A. Joint UAV placement optimization, resource allocation, and computation offloading for THz band: A DRL approach. *IEEE Trans. Wirel. Commun.* **22**(7), 4890–4900. <https://doi.org/10.1109/TWC.2022.3230407> (2023).
20. Mei, H., Yang, K., Liu, Q. & Wang, K. 3D-trajectory and phase-shift design for RIS-assisted UAV systems using deep reinforcement learning. *IEEE Trans. Veh. Technol.* **71**(3), 3020–3029. <https://doi.org/10.1109/TVT.2022.3143839> (2022).
21. Zhou, Y. et al. Uplink performance analysis of RIS-assisted UAV communication systems with random 3-D mobile pattern. *IEEE Trans. Commun.* **72**(7), 4288–4303. <https://doi.org/10.1109/TCOMM.2024.3391827> (2024).
22. De Alwis, C. et al. Survey on 6G frontiers: Trends, applications, requirements, technologies and future research. *IEEE Open J. Commun. Soc.* **2**, 836–886 (2021).
23. Mahmoud, A. et al. Intelligent reflecting surfaces assisted UAV communications for IoT networks: Performance analysis. *IEEE Trans. Green Commun. Network.* **5**(2), 1–1. <https://doi.org/10.1109/TGCN.2021.3068739> (2021).
24. Wang, D. et al. Passive beamforming and trajectory optimization for reconfigurable intelligent surface-assisted UAV secure communication. *Remote Sens.* **13**(21), 4286. <https://doi.org/10.3390/rs13214286> (2021).
25. Cui, T. J., Qi, M. Q., Wan, X., Zhao, J. & Cheng, Q. Coding metamaterials, digital metamaterials and programmable metamaterials. *Light Sci. Appl.* **3**(10), e218 (2014).
26. Huang, C., Zappone, A., Alexandropoulos, G. C., Debbah, M. & Yuen, C. Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends. *IEEE Wirel. Commun.* **27**(5), 118–125 (2020).
27. Alfattani, S., Yadav, A., Yanikomeroglu, H. & Yongacoglu, A. Beyond-cell communications via HAPS-RIS. In *Proc. IEEE Global Communications Conference (GLOBECOM), Rio de Janeiro, Brazil*, 3558–3563 (2022).
28. Dao, N.-N. et al. Survey on aerial radio access networks: Toward a comprehensive 6G access infrastructure. *IEEE Commun. Surv. Tutor.* **23**(2), 1193–1225 (2021).
29. Myyara, M., Lagnfdi, O., Darif, A. & Farchane, A. Enhancing QoS for IoT devices through heuristics-based computation offloading in multi-access edge computing. *Infocommun. J.* **XVII**(4), 12–20 (2024).
30. Manlove, D. F. *Algorithmics of matching under preferences* (World Scientific, 2013).
31. Yang, L. et al. Multi-UAV-enabled load-balance mobile-edge computing for IoT networks. *IEEE Internet Things J.* **7**(8), 6898–6908. <https://doi.org/10.1109/JIOT.2020.2971645> (2020).
32. Ren, Q., Abbasi, O., Kurt, G. K., Yanikomeroglu, H. & Chen, J. Caching and computation offloading in high altitude platform station (HAPS) assisted intelligent transportation systems. (2021).
33. Zhang, L., Zhang, H., Guo, C., Xu, H., Song, L. & Han, Z. Satellite-aerial integrated computing in disasters: User association and offloading decision. In *IEEE International Conference on Communications (ICC)*, Dublin, Ireland, pp. 554–559 (2020).
34. Yang, Z., Pan, C., Wang, K. & Shikh-Bahaei, M. Energy efficient resource allocation in UAV-enabled mobile edge computing networks. *IEEE Trans. Wirel. Commun.* **18**(9), 4576–4589 (2019).

Author contributions

B.D. produced the original article and ran the necessary simulations of the RIS-enhanced hierarchical aerial computing system, as well as the RCG algorithm implementation and performance analysis. I.I.I. and A.M.A. helped develop the approach and oversaw the suggested work. M.M.A. examined and edited the manuscript. All authors have reviewed and approved the final version of the manuscript.

Funding

Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to B.D.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2026