

# Hierarchical Trajectory and Offloading Optimization for Task Delay Minimization in UAV-Assisted MEC

Ruiyi Yao<sup>1</sup>, Weiyi Hu<sup>1</sup>, Leheng Qi, Xiaoyang Li<sup>2</sup>, Zidong Han, Yi Gong  
Southern University of Science and Technology, Guangdong, China

**Abstract**—This paper addresses the minimization problem of long-term end-to-end delay within a heterogeneous three-tier unmanned aerial vehicle (UAV)-assisted mobile edge computing network featuring two-hop forwarding. The joint optimization of sensor-node–aggregator association, access/backhaul bandwidth allocation, UAV trajectories, and offloading ratios is formulated as a mixed-integer nonlinear stochastic program characterized by complex cross-timeslot queue coupling. By exploiting the inherent asymmetric coupling across network tiers, a weighted weak-coupling surrogate with a provable layering-gap bound is derived. Based on this surrogate, a hierarchical online controller is designed, comprising a queue-aware collection layer, a backhaul coordination layer, and an offloading layer combining closed-form equalization with residual proximal policy optimization. Simulation results validate the effectiveness of the proposed framework, demonstrating significant delay reductions compared to state-of-the-art baselines.

**Index Terms**—Low-altitude wireless networks, unmanned aerial vehicle, mobile edge computing, hierarchical control, data offloading.

## I. INTRODUCTION

Emerging applications in low-altitude wireless networks (LAWN)—spanning environmental monitoring to disaster response—impose heterogeneous sensing, communication, and computation demands on unmanned aerial vehicles (UAVs), many of which are inherently time-critical [1], [2]. However, the size, weight, and power constraints of UAVs lead to limited computing capability [3], while offloading to remote clouds incurs prohibitive latencies [4]. Mobile edge computing (MEC) addresses this by migrating computation to the network frontier [5], [6], making it essential for LAWN deployments.

A single-tier deployment fails to reconcile wide-area coverage with localized computing: High-performance onboard processors are energy-prohibitive, yet lightweight drones saturate at traffic hotspots [7]. This motivates the heterogeneous three-tier architecture in Fig. 1: (i) ground-based sensor nodes (SNs) generate time-sensitive data; (ii) sensing aggregators (SAs) collect data and perform local processing; and (iii) a computing hub (CH) provides intensive offloaded computation. The resulting two-hop framework (SN-to-SA collection followed by SA-to-CH offloading) aligns heterogeneous capabilities with fluctuating traffic demands.

However, the performance of this architecture hinges on the tight coupling among user association, trajectory design,

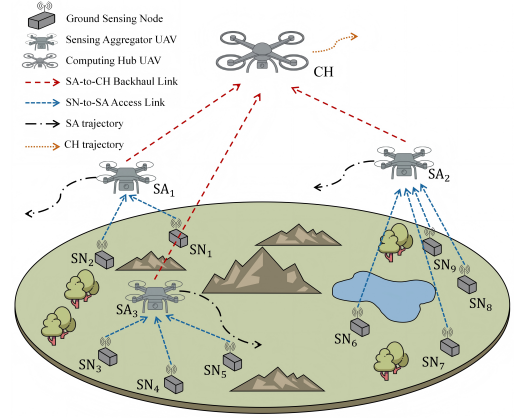


Fig. 1. Heterogeneous three-tier UAV-assisted MEC scenario.

and cross-layer resource control. Since queues at the SNs, SAs, and CH evolve interdependently, any single-tier configuration propagates throughout the network, necessitating joint orchestration of all decisions.

Existing work is insufficient for this interplay. Computation partitioning in [3] assumes single-hop topology, leaving the backhaul bottleneck unaddressed. Joint trajectory–resource methods in [8]–[10] neglect queue-state feedback, causing buffer overflows under bursty traffic. Lyapunov-based approaches in [11]–[13] optimize only variable subsets while fixing trajectories or association.

To address these challenges, this paper investigates the long-term end-to-end (E2E) delay minimization problem for the proposed three-tier UAV-assisted MEC system. The joint optimization of user association, trajectory design, and radio resource allocation is formulated as a mixed-integer nonlinear program. By exploiting the asymmetric coupling inherent across the network tiers, a weighted weak-coupling surrogate with a provable layering-gap bound is derived. This allows us to decompose the original problem into three sub-problems: (i) a queue-aware SN-to-SA collection sub-problem for user association and bandwidth optimization; (ii) a backhaul coordination sub-problem for trajectory planning; and (iii) an offloading sub-problem that combines a closed-form Equalize rule with residual proximal policy optimization (PPO). Numerical results validate our approach, demonstrating that the proposed method significantly outperforms state-of-the-art baselines for task delay minimization.

<sup>1</sup>These authors contributed equally to this work.

<sup>2</sup>Corresponding Author: Xiaoyang Li (Email:lixxy@sustech.edu.cn)

## II. SYSTEM MODEL AND PROBLEM FORMULATION

### A. System Architecture and Delay Model

A heterogeneous three-tier UAV-assisted MEC network with  $M$  SNs,  $K$  SAs, and one CH over an  $L \times L$  area is considered. Let  $\mathcal{M} \triangleq \{1, \dots, M\}$  and  $\mathcal{K} \triangleq \{1, \dots, K\}$  denote the SN and SA index sets, respectively. Time is equally partitioned into slots of duration  $\tau$ . As shown in Fig. 1, SNs upload data to their associated SAs, and each SA partitions its computing workload between local execution and CH offloading. In slot  $t$ , the control variables are

- Association:  $\mathcal{A}(t) \triangleq \{a_{m,k}(t) \in \{0, 1\}\}_{m \in \mathcal{M}, k \in \mathcal{K}}$ , where  $a_{m,k}(t) = 1$  if and only if SN  $m$  is served by SA  $k$ , with each SN associated with at most one SA:  $\sum_{k=1}^K a_{m,k}(t) \leq 1, \forall m$ ;
- Bandwidth:  $\mathcal{W}(t) \triangleq \{w_{m,k}(t)\}_{m,k} \cup \{w_k^{\text{CH}}(t)\}_k$ , representing the SN-SA and SA-CH bandwidth allocations;
- Positions:  $\mathcal{P}(t) \triangleq \{\mathbf{p}_k(t)\}_{k=1}^K \cup \{\mathbf{p}_{\text{CH}}(t)\}$ , subject to the maximum UAV speed  $v_{\max}$  for each SA and the CH, i.e.,  $\|\mathbf{u}_k(t+1) - \mathbf{u}_k(t)\| \leq v_{\max}\tau, \forall k, \|\mathbf{u}_{\text{CH}}(t+1) - \mathbf{u}_{\text{CH}}(t)\| \leq v_{\max}\tau$ , and  $\mathbf{u}_k(t), \mathbf{u}_{\text{CH}}(t) \in [0, L]^2$ ;
- Offloading ratios:  $\boldsymbol{\eta}(t) \triangleq \{\eta_k(t) \in [0, 1]\}_{k=1}^K$ .

For SA  $k$ , let  $\mathcal{M}_k(t) = \{m \in \mathcal{M} : a_{m,k}(t) = 1\}$  denote its associated SN set. Let  $\mathbf{u}_k(t)$  and  $\mathbf{u}_{\text{CH}}(t)$  denote horizontal UAV coordinates, with positions  $\mathbf{p}_k(t) = [\mathbf{u}_k^\top(t), h_k]^\top$  and  $\mathbf{p}_{\text{CH}}(t) = [\mathbf{u}_{\text{CH}}^\top(t), h_{\text{CH}}]^\top$ , and SN position  $\mathbf{q}_m = [x_m, y_m]^\top$ . All tasks are assumed to share a homogeneous computation profile requiring  $\nu$  central processing unit (CPU) cycles per bit. Accordingly, the effective computing rates (bits/s) are  $C_{\text{local}} = f_{\text{local}}/\nu$  and  $C_{\text{CH}} = f_{\text{CH}}/\nu$ , where  $f_{\text{local}}$  and  $f_{\text{CH}}$  denote the SA and CH CPU frequencies (cycles/s), respectively.

Let  $A_k(t) \in [0, R_{1,k}(t)\tau]$  denote the admitted first-hop collection batch (bits) entering SA  $k$ ,  $B_k(t)$  denote the SA data backlog (bits), and  $Q_{\text{CH}}(t)$  denote the CH data backlog (bits). The full backlog  $\tilde{B}_k(t)$  is distinguished from the scheduled processing batch  $B_k(t) \in [0, B_k(t)]$ , which represents the amount of data that SA  $k$  can process within slot  $t$ :

$$\tilde{B}_k(t) = \min\{B_k(t), (C_{\text{local}} + R_{2,k}(t))\tau\}, \quad (1)$$

where  $R_{1,k}(t)$  and  $R_{2,k}(t)$  denote the aggregate SN-to-SA collection rate and the SA-to-CH backhaul transmission rate, respectively (both defined in Sub-section B).

The E2E delay for sensing aggregator (SA)  $k$  in slot  $t$  is defined as the total latency from the beginning of data collection at the SNs to the completion of all associated computation tasks. It includes the delays of SN-to-SA transmission, SA-to-CH offloading, and concurrent workload execution at the SA and the CH. Thus, the E2E delay can be expressed as

$$D_k^{\text{e2e}}(t) = t_{1,k}(t) + \max(t_{3,k}(t), t_{2,k}(t) + t_{4,k}(t)), \quad (2)$$

where the four delay components—SN-to-SA collection (3a), SA-to-CH backhaul (3b), SA local computing (3c), and CH remote computing (3d)—are defined as

$$t_{1,k}(t) = A_k(t) / R_{1,k}(t), \quad (3a)$$

$$t_{2,k}(t) = \eta_k(t) \tilde{B}_k(t) / R_{2,k}(t), \quad (3b)$$

$$t_{3,k}(t) = (1 - \eta_k(t)) \tilde{B}_k(t) / C_{\text{local}}, \quad (3c)$$

$$t_{4,k}(t) = (Q_{\text{CH}}(t) + \eta_k(t) \tilde{B}_k(t)) / C_{\text{CH}}. \quad (3d)$$

For notational compactness, any ratio with a zero numerator is defined as zero. Under the feasibility constraint (17),  $R_{2,k}(t) = 0$  implies  $\eta_k(t) \tilde{B}_k(t) = 0$ ; hence,  $t_{2,k}(t)$  is well-defined.

**Remark 1.** The  $\max(\cdot)$  in (2) is non-smooth and non-separable, precluding per-layer closed-form solutions and drift-plus-penalty decomposition. We therefore replace  $\max(t_{3,k}, t_{2,k} + t_{4,k})$  with a weighted surrogate  $\alpha t_{2,k} + \beta t_{3,k} + \gamma t_{4,k}$  in Sub-section II-D. This is justified because: (i) only the objective is modified, so every surrogate-feasible solution remains feasible for  $D_k^{\text{e2e}}(t)$ ; (ii) the sub-optimality incurred by solving the surrogate in a layered fashion is  $\mathcal{O}(\varepsilon)$  with  $\varepsilon \triangleq \alpha + \beta + \gamma \ll 1$ , vanishing as  $\varepsilon \rightarrow 0$  (Proposition 1, Section III); and (iii) the weighted-sum form separates additively across hops, enabling the layered design in Section III.

### B. Communication Model

Shannon rates are adopted with a probabilistic line-of-sight (LoS)/non-LoS (NLoS) air-to-ground model for the SN-SA link and a dominant-LoS air-to-air model for the SA-CH link. The SN-SA and SA-CH distances are

$$d_{m,k}(t) = \sqrt{\|\mathbf{u}_k(t) - \mathbf{q}_m\|^2 + h_k^2}, \quad (4)$$

$$d_{k,\text{CH}}(t) = \sqrt{\|\mathbf{u}_k(t) - \mathbf{u}_{\text{CH}}(t)\|^2 + (h_{\text{CH}} - h_k)^2}.$$

Let the elevation angle from SN  $m$  to SA  $k$  be

$$\theta_{m,k}(t) = \frac{180}{\pi} \arcsin\left(\frac{h_k}{d_{m,k}(t)}\right), \quad (5)$$

and define the LoS probability by the standard sigmoid model

$$P_{m,k}^{\text{LoS}}(t) = \frac{1}{1 + a_{\text{env}} \exp(-b_{\text{env}}(\theta_{m,k}(t) - a_{\text{env}}))}, \quad (6)$$

where  $a_{\text{env}}$  and  $b_{\text{env}}$  are environment-dependent parameters. The corresponding average large-scale SN-SA channel gain is modeled as

$$\bar{g}_{m,k}(t) = P_{m,k}^{\text{LoS}}(t) \beta_{\text{L}} d_{m,k}(t)^{-\alpha_{\text{L}}} + (1 - P_{m,k}^{\text{LoS}}(t)) \beta_{\text{N}} d_{m,k}(t)^{-\alpha_{\text{N}}}, \quad (7)$$

with  $\alpha_{\text{L}} < \alpha_{\text{N}}$  and  $\beta_{\text{L}} > \beta_{\text{N}}$ . For the SA-CH backhaul, a dominant-LoS air-to-air model is adopted. The unit-bandwidth spectral efficiencies are

$$r_{m,k}^{\text{SN-SA}}(t) = \log_2 \left( 1 + \frac{P_{\text{SN}} \bar{g}_{m,k}(t)}{N_0} \right), \quad (8)$$

$$r_{2,k}(t) = \log_2 \left( 1 + \frac{P_{\text{SA}} G_0 d_{k,\text{CH}}(t)^{-\alpha_{\text{A}}}}{N_0} \right).$$

The corresponding rates are

$$R_{m,k}^{\text{SN-SA}}(t) = w_{m,k}(t) r_{m,k}^{\text{SN-SA}}(t), \quad (9a)$$

$$R_{1,k}(t) = \sum_{m \in \mathcal{M}_k(t)} R_{m,k}^{\text{SN-SA}}(t), \quad (9b)$$

$$R_{2,k}(t) = w_k^{\text{CH}}(t) r_{2,k}(t), \quad (9c)$$

where  $\sum_{m \in \mathcal{M}_k(t)} w_{m,k}(t) \leq W_k$  for each SA  $k$  and  $\sum_{k=1}^K w_k^{\text{CH}}(t) \leq W^{\text{CH}}$  for the CH.

### C. Queue Dynamics and Stability Conditions

Let  $n_m(t)$  denote the exogenous data arrivals (bits) at SN  $m$  in slot  $t$ , which follow a Poisson distribution with mean  $\bar{n}_m = \lambda_m s_{\text{task}}\tau$ , where  $\lambda_m$  is the task arrival rate (tasks/s)

and  $s_{\text{task}}$  is the per-task data size (bits). The SN, SA, and CH queues evolve as

$$s_m(t) \triangleq \sum_{k=1}^K a_{m,k}(t) R_{m,k}^{\text{SN-SA}}(t) \tau, \quad (10)$$

$$Q_m^{\text{SN}}(t+1) = \max[Q_m^{\text{SN}}(t) - s_m(t), 0] + n_m(t), \quad (11)$$

$$B_k(t+1) = \max[B_k(t) - \mu_k(t)\tau, 0] + A_k(t), \quad (12)$$

$$\mu_k(t) = (1 - \eta_k(t))C_{\text{local}} + \eta_k(t)R_{2,k}(t), \quad (13)$$

$$A_k^{\text{CH}}(t) \triangleq \eta_k(t)\tilde{B}_k(t), \quad (14)$$

$$Q_{\text{CH}}(t+1) = \max[Q_{\text{CH}}(t) - C_{\text{CH}}\tau, 0] + \sum_{k=1}^K A_k^{\text{CH}}(t). \quad (15)$$

The batch-based delay model requires the one-slot feasibility conditions

$$(1 - \eta_k(t))\tilde{B}_k(t) \leq C_{\text{local}}\tau, \quad (16)$$

$$\eta_k(t)\tilde{B}_k(t) \leq R_{2,k}(t)\tau. \quad (17)$$

Mean-rate stability is adopted as the queue-stability criterion, i.e.,

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E}[X(T)]}{T} = 0, \quad X \in \{Q_m^{\text{SN}}, B_k, Q_{\text{CH}}\}, \quad \forall m, k. \quad (18)$$

#### D. Long-Term Optimization Problem

As motivated in Remark 1, the  $\max(\cdot)$  operator in  $D_k^{\text{e2e}}(t)$  is replaced by a weighted linear surrogate:

$$D_k(t) \triangleq t_{1,k}(t) + \alpha t_{2,k}(t) + \beta t_{3,k}(t) + \gamma t_{4,k}(t), \quad (19)$$

where  $\alpha, \beta, \gamma \geq 0$  are small modeling weights satisfying  $\varepsilon \triangleq \alpha + \beta + \gamma \ll 1$ , so the downstream terms act as perturbative corrections to the dominant first-hop term.

Define the long-term average surrogate delay as

$$\bar{D} \triangleq \limsup_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E} \left[ \frac{1}{|\mathcal{K}_{\text{act}}(t)|} \sum_{k \in \mathcal{K}_{\text{act}}(t)} D_k(t) \right], \quad (20)$$

where  $\mathcal{K}_{\text{act}}(t) \triangleq \{k \in \mathcal{K} : \tilde{B}_k(t) > 0\}$  is the set of SAs with non-zero scheduled workload in slot  $t$ . The per-slot average is defined as zero when  $\mathcal{K}_{\text{act}}(t) = \emptyset$ .

Let  $\Omega_{\text{std}}(t)$  collect the per-slot constraints introduced in Sub-sections A–C: single-SA association ( $\sum_k a_{m,k}(t) \leq 1$ ,  $a_{m,k}(t) \in \{0, 1\}$ ), SA/CH bandwidth budgets ( $\sum_m w_{m,k}(t) \leq W_k$ ,  $\sum_k w_k^{\text{CH}}(t) \leq W^{\text{CH}}$ ), speed and service-area limits ( $\|\mathbf{u}_k(t+1) - \mathbf{u}_k(t)\| \leq v_{\text{max}}\tau$ ,  $\forall k$ ,  $\|\mathbf{u}_{\text{CH}}(t+1) - \mathbf{u}_{\text{CH}}(t)\| \leq v_{\text{max}}\tau$ , and  $\mathbf{u}_k(t), \mathbf{u}_{\text{CH}}(t) \in [0, L]^2$ ), and offloading constraints  $\eta_k(t) \in [0, 1]$ ;  $\tilde{B}_k(t)$  is given by (1). The long-term delay minimization problem is formulated as

$$(\mathbf{P}_0) \quad \min_{\mathcal{A}(t), \mathcal{W}(t), \mathcal{P}(t), \boldsymbol{\eta}(t)} \bar{D} \quad (21a)$$

$$\text{s.t.} \quad (\mathcal{A}(t), \mathcal{W}(t), \mathcal{P}(t), \boldsymbol{\eta}(t)) \in \Omega_{\text{std}}(t), \forall t, \quad (21b)$$

$$(1 - \eta_k(t))\tilde{B}_k(t) \leq C_{\text{local}}\tau, \quad (21c)$$

$$\eta_k(t)\tilde{B}_k(t) \leq R_{2,k}(t)\tau. \quad (21d)$$

Constraints (21c) and (21d) hold for all  $k$  and  $t$ . Queue stability is enforced by (18). Problem  $\mathbf{P}_0$  is a mixed-integer nonlinear stochastic program with cross-timeslot coupling through the queue dynamics (11)–(15), which motivates the hierarchical online method developed next.

### III. PROPOSED METHODOLOGY

Since  $\mathbf{P}_0$  involves integer association variables and is NP-hard even for a single slot, direct solution is intractable. A key observation is the *asymmetric coupling* in (19): The dominant first-hop delay  $t_{1,k}$ , carrying unit weight, depends solely on the collection-layer variables ( $\mathcal{A}(t), \{w_{m,k}(t)\}, \{\mathbf{p}_k(t)\}$ ), whereas the downstream terms  $t_{2,k} - t_{4,k}$  additionally involve backhaul and offloading variables ( $\eta_k(t), \{w_k^{\text{CH}}(t)\}, \mathbf{p}_{\text{CH}}(t)$ ) yet carry only a combined weight of  $\varepsilon \ll 1$ . This weight disparity permits the collection layer to be optimized independently with bounded sub-optimality. Sub-section A formalizes this via a weak-coupling layering-gap bound, and Sub-section B decomposes  $\mathbf{P}_0$  into three sub-problems: collection ( $\mathbf{P}_1$ ), backhaul ( $\mathbf{P}_2$ ), and offloading ( $\mathbf{P}_3$ ) with a closed-form Equalize rule refined by residual PPO. Accordingly, a hierarchical online controller is constructed with collection, backhaul, and offloading layers. Sub-sections C–F detail the algorithmic solutions.

#### A. Weighted Weak-Coupling Hierarchical Modeling

The per-slot decision is partitioned as  $\mathbf{x} \triangleq (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ , where  $\mathbf{x}_1 \triangleq (\mathcal{A}(t), \{w_{m,k}(t)\}, \{\mathbf{p}_k(t)\})$  (collection),  $\mathbf{x}_2 \triangleq (\{w_k^{\text{CH}}(t)\}, \mathbf{p}_{\text{CH}}(t))$  (backhaul), and  $\mathbf{x}_3 \triangleq (\{\eta_k(t)\})$  (offloading). Let  $J(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$  denote the single-slot expected surrogate cost induced by (19) under  $\mathbf{x}$ . Under the weak-coupling condition  $\varepsilon \triangleq \alpha + \beta + \gamma \ll 1$ , this cost can be decomposed as

$$J(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = J_1(\mathbf{x}_1) + J_{\text{coup}}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3), \quad (22)$$

where  $J_1(\mathbf{x}_1) \triangleq \frac{1}{|\mathcal{K}_{\text{act}}(t)|} \sum_{k \in \mathcal{K}_{\text{act}}(t)} \mathbb{E}[t_{1,k}(t)]$  is the expected average first-hop delay, which depends only on collection-layer decisions, and  $J_{\text{coup}} \triangleq J - J_1$  collects the  $\mathcal{O}(\varepsilon)$  weighted downstream coupling.

Let  $\mathcal{X}$  denote the feasible set of  $\mathbf{P}_0$ , and define the projection onto the collection-layer (first-tier) decisions:

$$\mathcal{X}_1^{\text{proj}} \triangleq \{\mathbf{x}_1 : \exists (\mathbf{x}_2, \mathbf{x}_3), (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \in \mathcal{X}\}, \quad (23)$$

as well as the downstream feasible slice conditioned on a given collection-layer decision:

$$\mathcal{X}_{23}(\mathbf{x}_1) \triangleq \{(\mathbf{x}_2, \mathbf{x}_3) : (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \in \mathcal{X}\}. \quad (24)$$

A *layered* (i.e., separated) benchmark first solves  $\mathbf{x}_1^{\text{sep}} \in \arg \min_{\mathbf{x}_1 \in \mathcal{X}_1^{\text{proj}}} J_1(\mathbf{x}_1)$  independently, then optimizes  $(\mathbf{x}_2^{\text{sep}}, \mathbf{x}_3^{\text{sep}}) \in \arg \min_{(\mathbf{x}_2, \mathbf{x}_3) \in \mathcal{X}_{23}(\mathbf{x}_1^{\text{sep}})} J_{\text{coup}}(\mathbf{x}_1^{\text{sep}}, \mathbf{x}_2, \mathbf{x}_3)$ .

To quantify the sub-optimality incurred by solving  $\mathbf{P}_0$  in a layered fashion rather than jointly, the following assumption is introduced.

**Assumption A1 (Uniform Backlog Bounds):** Finite buffers or explicit state truncation ensure  $0 \leq B_k(t) \leq B_{\text{max}}$  and  $0 \leq Q_{\text{CH}}(t) \leq Q_{\text{max}}^{\text{CH}}$  for all  $k$  and  $t$ .

Assumption A1 is consistent with the bounded-backlog assumptions used in prior Lyapunov-based optimization studies [11], [12] and with practical UAV buffer limits. It implies bounded downstream delays and leads to the following gap certificate.

**Proposition 1 (Weak-Coupling Layering Gap):** Under Assumption A1 and the existence of a feasible stationary randomized policy for  $\mathbf{P}_0$ ,

$$|J^* - J_{\text{sep}}^*| \leq \Delta_{\text{coup}} \triangleq \alpha\tau + \beta\tau + \gamma \frac{Q_{\text{max}}^{\text{CH}} + B_{\text{max}}}{C_{\text{CH}}}, \quad (25)$$

where  $J^*$  and  $J_{\text{sep}}^*$  are the joint and layered optima, respectively, and  $\Delta_{\text{coup}} \rightarrow 0$  as  $\varepsilon \rightarrow 0$ .

Proof: Let  $J_1^* \triangleq \min_{\mathbf{x}_1 \in \mathcal{X}_1^{\text{proj}}} J_1(\mathbf{x}_1)$ . Both  $J^*$  and  $J_{\text{sep}}^*$  lie in  $[J_1^*, J_1^* + \Delta_{\text{coup}}]$ , which follows from the one-slot feasibility constraints and Assumption A1.  $\square$

### B. Problem Decomposition

Based on the layering-gap bound established above, the hierarchical online controller decomposes  $\mathbf{P}_0$  into three sub-problems:

$$(\mathbf{P}_1) \quad \min_{\mathbf{x}_1 \in \mathcal{X}_1^{\text{proj}}} J_1(\mathbf{x}_1), \quad (26)$$

for SN-to-SA collection;

$$(\mathbf{P}_2) \quad \max_{\{w_k^{\text{CH}}(t)\}, \mathbf{p}_{\text{CH}}(t)} \Phi_2(t), \quad (27)$$

for SA-to-CH backhaul coordination, subject to the CH bandwidth budget and speed limits, where the utility  $\Phi_2$  is defined in Sub-section D; and

$$(\mathbf{P}_3) \quad \min_{\{\eta_k(t)\}} \limsup_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E} \left[ \frac{1}{|\mathcal{K}_{\text{act}}(t)|} \sum_{k \in \mathcal{K}_{\text{act}}(t)} g_k(t) \right], \quad (28)$$

for queue-aware offloading, where the normalized term is defined as zero when  $\mathcal{K}_{\text{act}}(t) = \emptyset$ , and

$$g_k(t) \triangleq \alpha t_{2,k}(t) + \beta t_{3,k}(t) + \gamma t_{4,k}(t). \quad (29)$$

Because  $J_1$  is a single-slot objective,  $\mathbf{P}_1$  and  $\mathbf{P}_2$  are solved independently in each slot, whereas the cross-timeslot queue coupling in  $g_k$  requires the Lyapunov-based long-term formulation of  $\mathbf{P}_3$ .

### C. $\mathbf{P}_1$ : SN-to-SA Collection Control

To solve  $\mathbf{P}_1$ , a queue-aware association and bandwidth-allocation rule is adopted. If SN  $m$  were assigned to SA  $k$ , the corresponding predicted upload completion time is

$$\zeta_{m,k}(t) = \frac{\sum_{n \in \mathcal{M}_k(t)} Q_n^{\text{SN}}(t)/r_{n,k}^{\text{SN-SA}}(t)}{W_k} + \frac{Q_m^{\text{SN}}(t)/r_{m,k}^{\text{SN-SA}}(t)}{W_k}, \quad (30)$$

The association decision is then obtained greedily, with  $\mathcal{M}_k(t)$  updated after each assignment, as

$$k_m^*(t) = \arg \min_k \zeta_{m,k}(t). \quad (31)$$

Given the greedy association obtained from (31), the bandwidth allocated to SN  $m$  at SA  $k$  is

$$w_{m,k}^*(t) = W_k \cdot \frac{Q_m^{\text{SN}}(t)/r_{m,k}^{\text{SN-SA}}(t)}{\sum_{n \in \mathcal{M}_k(t)} Q_n^{\text{SN}}(t)/r_{n,k}^{\text{SN-SA}}(t)}. \quad (32)$$

The SA trajectory is updated via projected gradient descent on  $t_{1,k}$  with respect to  $\mathbf{u}_k(t)$ , where the displacement is projected onto  $\|\Delta \mathbf{u}_k\| \leq v_{\text{max}}\tau$ .

### D. $\mathbf{P}_2$ : SA-to-CH Backhaul and CH Mobility Control

To solve  $\mathbf{P}_2$ , a virtual backhaul queue is introduced:

$$U_k(t+1) = \max[U_k(t) - s_k^{\text{CH}}(t), 0] + s_k^{\text{SA}}(t), \quad (33)$$

where  $U_k(1) = 0$ ,  $s_k^{\text{CH}}(t) = w_k^{\text{CH}}(t)r_{2,k}(t)\tau$ , and  $s_k^{\text{SA}}(t) \triangleq \hat{\eta}_k(t)\tilde{B}_k(t)$ . The predictor is set as  $\hat{\eta}_k(t) = \eta_k(t-1)$  with  $\hat{\eta}_k(1) = 0$ . This choice is adopted because  $\mathbf{P}_2$  is executed before  $\mathbf{P}_3$  in each slot, so  $\eta_k(t)$  is not yet available. To equalize backhaul completion times across all SAs, the CH bandwidth is allocated in proportion to the pending workload normalized by spectral efficiency:

$$w_k^{\text{CH},*}(t) = W^{\text{CH}} \cdot \frac{U_k(t)/r_{2,k}(t)}{\sum_{j=1}^K U_j(t)/r_{2,j}(t)}. \quad (34)$$

The backhaul utility in (27) is then defined as

$$\Phi_2(t) \triangleq - \max_k \frac{U_k(t)}{w_k^{\text{CH}}(t)r_{2,k}(t)}. \quad (35)$$

With  $w_k^{\text{CH}}(t)$  determined by (34),  $\mathbf{P}_2$  reduces to optimizing the CH trajectory  $\mathbf{u}_{\text{CH}}(t)$ . The CH position is updated via projected gradient descent on  $\sum_{k=1}^K U_k(t)/r_{2,k}(t)$  with respect to  $\mathbf{u}_{\text{CH}}(t)$ , subject to  $\|\Delta \mathbf{u}_{\text{CH}}\| \leq v_{\text{max}}\tau$ . Under (34), this objective is equivalent to maximizing the utility  $\Phi_2(t)$  in (35).

### E. $\mathbf{P}_3$ : Lyapunov-Based Queue-Aware Offloading

After  $\mathbf{P}_1$  and  $\mathbf{P}_2$  are solved, the offloading layer acts on the SA backlogs  $\{B_k(t)\}$  and CH backlog  $Q_{\text{CH}}(t)$ . Define the queue state as

$$\Theta(t) \triangleq (B_1(t), \dots, B_K(t), Q_{\text{CH}}(t)), \quad (36)$$

and the quadratic Lyapunov function as

$$L(\Theta(t)) \triangleq \frac{1}{2} \left[ \sum_{k=1}^K B_k^2(t) + Q_{\text{CH}}^2(t) \right]. \quad (37)$$

The conditional Lyapunov drift is defined as  $\Delta L(\Theta(t)) \triangleq \mathbb{E}[L(\Theta(t+1)) - L(\Theta(t)) \mid \Theta(t)]$ , where the conditional expectation is taken with respect to the current queue state  $\Theta(t)$ . Using (13), (14), and the queue dynamics in (12) and (15), and applying  $(\max\{x-y, 0\}+z)^2 \leq x^2+y^2+z^2-2x(y-z)$ , the drift is upper-bounded as

$$\begin{aligned} \Delta L(\Theta(t)) &\leq C_0 - \sum_{k=1}^K B_k(t) \mathbb{E}[\mu_k(t)\tau - A_k(t) \mid \Theta(t)] \\ &\quad - Q_{\text{CH}}(t) \mathbb{E} \left[ C_{\text{CH}}\tau - \sum_{k=1}^K A_k^{\text{CH}}(t) \mid \Theta(t) \right], \end{aligned} \quad (38)$$

where  $C_0 > 0$  is a finite constant collecting the quadratic arrival and service terms.

To jointly penalize delay and congestion, define the drift-plus-penalty quantity as

$$\text{DPP}(t) \triangleq \Delta L(\Theta(t)) + \frac{V}{|\mathcal{K}_{\text{act}}(t)|} \mathbb{E} \left[ \sum_{k \in \mathcal{K}_{\text{act}}(t)} g_k(t) \mid \Theta(t) \right], \quad (39)$$

where  $V > 0$  controls the delay-backlog trade-off, and the normalized penalty term is again taken as zero when  $\mathcal{K}_{\text{act}}(t) = \emptyset$ . After  $\mathbf{P}_1$  and  $\mathbf{P}_2$  are solved,  $A_k(t)$  and  $R_{2,k}(t)$  are fixed for the current slot. Therefore, substituting the upper

bound (38) into (39) and discarding terms independent of  $\{\eta_k(t)\}$  reduces the offloading decision to the following per-slot problem:

$$\begin{aligned}
(\mathbf{P}_{3\text{-slot}}) \quad & \min_{\{\eta_k(t)\}} \sum_{k=1}^K B_k(t) [C_{\text{local}} - R_{2,k}(t)] \eta_k(t) \tau \\
& + \sum_{k=1}^K Q_{\text{CH}}(t) A_k^{\text{CH}}(t) \\
& + \frac{V}{|\mathcal{K}_{\text{act}}(t)|} \sum_{k \in \mathcal{K}_{\text{act}}(t)} g_k(t)
\end{aligned} \tag{40}$$

subject to  $0 \leq \eta_k(t) \leq 1$  and (21c)–(21d). According to standard Lyapunov optimization theory [11], [12], if a stationary randomized policy with positive slack exists and arrivals and service rates are bounded, exact per-slot minimization guarantees mean-rate stability and the standard delay–backlog trade-off: The optimality gap in  $\bar{D}$  is  $\mathcal{O}(1/V)$  while the average queue backlog grows as  $\mathcal{O}(V)$ .

#### F. Closed-Form Equalize Rule and Residual Learning Enhancement

Solving  $\mathbf{P}_{3\text{-slot}}$  exactly requires numerical optimization at every slot. A lightweight closed-form alternative is obtained by dropping the backlog-linear terms (the first two summations in (40)), which are negligible when queues are moderate relative to the penalty weight  $V$ , and equating the weighted local branch delay with the weighted offload branch delay, i.e.,  $\beta t_{3,k} = \alpha t_{2,k} + \gamma t_{4,k}$ , yielding the closed-form Equalize rule

$$\eta_k^{\text{eq}}(t) = \text{clip} \left( \frac{\beta/C_{\text{local}} - \gamma Q_{\text{CH}}(t)/(\tilde{B}_k(t) C_{\text{CH}})}{\beta/C_{\text{local}} + \alpha/R_{2,k}(t) + \gamma/C_{\text{CH}}}, 0, 1 \right). \tag{41}$$

For  $\tilde{B}_k(t) > 0$ , project  $\eta_k^{\text{eq}}(t)$  onto the feasible interval  $[\eta_k^{\min}(t), \eta_k^{\max}(t)]$ , where

$$\begin{aligned}
\eta_k^{\min}(t) &= \max\{0, 1 - C_{\text{local}}\tau/\tilde{B}_k(t)\}, \\
\eta_k^{\max}(t) &= \min\{1, R_{2,k}(t)\tau/\tilde{B}_k(t)\}.
\end{aligned} \tag{42}$$

If  $\tilde{B}_k(t) = 0$ , set  $\eta_k^{\text{eq}}(t) = 0$ . To capture multi-slot dynamics beyond the static Equalize rule, augment the analytical baseline as

$$\eta_k(t) = \text{clip}(\eta_k^{\text{eq}}(t) + s_\delta \delta_k(t), \eta_k^{\min}(t), \eta_k^{\max}(t)), \tag{43}$$

where  $\delta_k(t) \in [-1, 1]$  is the residual action and  $s_\delta = 0.3$  is its scaling factor. If  $\tilde{B}_k(t) = 0$ , set  $\eta_k(t) = 0$ .

The residual PPO agent uses a 37-dimensional observation (queue backlogs, rates, and computation capability) and a multi-layer perceptron (MLP) actor-critic (256-256-128, tanh). The reward is set as  $r(t) = -\bar{D}_{\text{slot}}(t) - \rho \frac{1}{K} \sum_k \delta_k^2(t)$  ( $\rho = 0.05$ ), where  $\bar{D}_{\text{slot}}(t)$  denotes the per-slot average of  $D_k(t)$  over  $\mathcal{K}_{\text{act}}(t)$ . Training is conducted via PPO [14] with Adam ( $10^{-4}$ ), discount factor 0.99, clip ratio  $\epsilon_c = 0.2$ , for 10M steps across 64 parallel environments.

TABLE I  
UNIFIED E2E PERFORMANCE COMPARISON ( $N = 100$  SEEDS)

| Policy                       | Surrogate $\bar{D}$ (s) | Mean E2E Delay (s) |
|------------------------------|-------------------------|--------------------|
| <b>Proposed-Hierarchical</b> | <b>0.221 ± 0.013</b>    | <b>3.29 ± 0.25</b> |
| TD3 [16]                     | 0.303 ± 0.021           | 6.91 ± 3.31        |
| DDPG [15]                    | 0.325 ± 0.028           | 7.34 ± 4.03        |
| SAC [17]                     | 0.431 ± 0.190           | 7.05 ± 3.87        |
| AO-Myopic [8], [9]           | 0.311 ± 0.021           | 7.70 ± 4.63        |
| Lyapunov [11], [12]          | 0.587 ± 0.072           | 4.48 ± 2.59        |
| Maritime-Lyapunov [18]       | 0.375 ± 0.024           | 10.03 ± 6.66       |
| Layered-Cost [19]            | 0.281 ± 0.019           | 5.66 ± 3.09        |
| Hier-MARL Proxy [13]         | 0.287 ± 0.020           | 6.35 ± 3.53        |

#### IV. PERFORMANCE EVALUATION

##### A. Simulation Setup

The simulation is conducted in a  $1000\text{m} \times 1000\text{m}$  area with  $M = 20$  SNs,  $K = 5$  SAs, and one CH. Poisson traffic with rate  $\lambda_m = 1.4$  tasks/s per SN arrives during the first 100 s (task size  $s_{\text{task}} = 15$  MB). The system processes the residual backlog until completion. Each SA's access bandwidth budget is 5 MHz and the shared CH backhaul budget is 40 MHz. The surrogate weights are  $\alpha = \beta = \gamma = 0.1$  ( $\varepsilon = 0.3$ , within the weak-coupling regime of Proposition 1) and the Lyapunov penalty weight is  $V = 50$ .

##### B. End-to-End Comparison Against Baselines

The proposed approach is benchmarked against direct reinforcement learning baselines, namely deep deterministic policy gradient (DDPG) [15], twin delayed DDPG (TD3) [16], and soft actor-critic (SAC) [17], as well as purpose-adapted heuristics, including AO-Myopic [8], [9], Lyapunov [11], [12], Maritime-Lyapunov [18], Layered-Cost [19], and Hier-MARL Proxy [13]. All baselines share the same  $\mathbf{P}_1/\mathbf{P}_2$  layers and differ only in  $\mathbf{P}_3$ . For fairness, the direct reinforcement learning baselines use the same network architecture and training budget (256-256-128, 10M steps, 64 environments).

As shown in Table I, the proposed controller achieves  $\bar{D} = 0.221$ , reducing the surrogate by 27.1% over TD3 (0.303) and 21.4% over Layered-Cost (0.281), while cutting mean E2E delay to 3.29 s versus 4.48–10.03 s for all competitors. Under a 10M-step training budget, the direct reinforcement learning baselines (TD3/DDPG/SAC) yield mean E2E delays of 6.91/7.34/7.05 s, all substantially higher than that of the proposed method. This behavior is consistent with the well-known difficulty of end-to-end optimization in high-dimensional, heterogeneous, discrete-continuous action spaces. These results suggest that hierarchical decomposition, when combined with residual learning, is more effective in reducing task delay.

##### C. Ablation Experiments

Two ablated variants—without  $\mathbf{P}_2$  (fixed CH, equal backhaul bandwidth) and without  $\mathbf{P}_1$  (nearest-SA, equal SA bandwidth)—are retrained from scratch under the same training setting as the full method, so that the contribution of each layer can be isolated.

Table II shows that removing  $\mathbf{P}_1$  degrades  $\bar{D}$  by 45.2% and E2E delay by 62.6%, whereas removing  $\mathbf{P}_2$  causes 16.3% and

TABLE II  
ABLATION RESULTS OF THE HIERARCHICAL CONTROLLER ( $N = 100$  SEEDS)

| Variant                | Surrogate $\bar{D}$ (s)             | Mean E2E Delay (s)                  |
|------------------------|-------------------------------------|-------------------------------------|
| <b>Full</b>            | <b><math>0.221 \pm 0.013</math></b> | <b><math>3.290 \pm 0.252</math></b> |
| Without $\mathbf{P}_2$ | $0.257 \pm 0.029$                   | $4.115 \pm 0.561$                   |
| Without $\mathbf{P}_1$ | $0.321 \pm 0.042$                   | $5.349 \pm 0.781$                   |

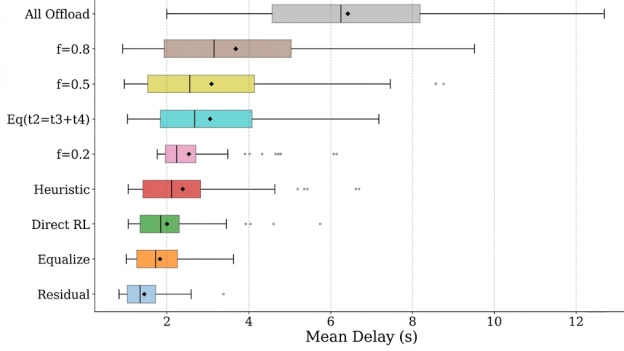


Fig. 2. Per-slot E2E delay distribution under  $\mathbf{P}_3$  variants with  $\mathbf{P}_1/\mathbf{P}_2$  fixed. “All-Local” omitted (much larger scale).

25.1% degradations, underscoring that queue-aware first-hop collection is the dominant source of gain.

With the outputs of  $\mathbf{P}_1$  and  $\mathbf{P}_2$  fixed, Fig. 2 compares  $\mathbf{P}_3$  variants in terms of per-slot E2E delay  $D_k^{\text{e2e}}(t)$ , including full offloading, fixed offloading fractions of 0.8, 0.5, and 0.2, an unweighted delay-equalizing heuristic ( $\text{Eq}(t_3 = t_2 + t_4)$ ), and PPO trained from scratch. Residual PPO achieves a mean per-slot delay of 1.44 s, compared with 1.83 s for Equalize alone and 2.00 s for PPO trained from scratch, which confirms the benefit of refining an analytical prior with residual learning.

#### D. Zero-Shot Robustness

The policy trained at the nominal  $1.0\times$  load is directly evaluated at multipliers  $0.8\times$ – $1.4\times$  without retraining. Mean E2E delay rises from 1.33 s to 16.32 s for the proposed controller at  $1.4\times$ , versus 27.05 s (Layered-Cost) and 29.81 s (TD3). The 95th-percentile E2E delay is reduced by 35.2% relative to Layered-Cost and 40.9% relative to TD3. Removing  $\mathbf{P}_1$  at  $1.4\times$  raises mean delay to 18.39 s, while removing  $\mathbf{P}_2$  has a minor effect (16.22 s), indicating that first-hop collection is the primary source of load-shift robustness.

#### V. CONCLUSION

A hierarchical controller for long-term delay minimization in a three-tier UAV-assisted MEC network has been presented. The performance evaluation shows consistent improvements over representative baselines, and the ablation experiments indicate that the queue-aware first-hop collection layer contributes the largest share of the overall gain. Future work will include multi-CH topologies and adaptive weight tuning.

#### ACKNOWLEDGEMENT

The work is supported in part by National Natural Science Foundation of China under Grant U25A20389, Guangdong 2026A1515010226, Science and Technology Program under Grant Shenzhen Science and Technology

Program under Grants KJZD20240903095402004 JCYJ20241202125328038, and JCYJ20241202124934046, and Young Elite Scientists Sponsorship Program by CAST under Grant YESS20240364

#### REFERENCES

- [1] Y. Jiang, X. Li, G. Zhu, H. Li, J. Deng, and K. Han, “Integrated sensing and communication for low altitude economy: Opportunities and challenges,” *IEEE Commun. Mag.*, vol. 63, no. 12, pp. 72–78, 2025.
- [2] Y. Jiang, X. Li, G. Zhu, K. Han, K. Meng, and W. Yang, “Network-level performance analysis for air-ground integrated sensing and communication,” *IEEE Trans. Wireless Commun.*, vol. 24, no. 8, pp. 6931–6946, 2025.
- [3] S. Wang, X. Li, and Y. Gong, “Energy-efficient task offloading and resource allocation for delay-constrained edge-cloud computing networks,” *IEEE Trans. Green Commun. Netw.*, vol. 8, no. 1, pp. 514–524, 2024.
- [4] C. You and K. Huang, “Exploiting non-causal cpu-state information for energy-efficient mobile cooperative computing,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4104–4117, 2018.
- [5] C. You, K. Huang, and H. Chae, “Energy efficient mobile cloud computing powered by wireless energy transfer,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, 2016.
- [6] Y. Zeng, R. Zhang, and T. J. Lim, “Wireless communications with unmanned aerial vehicles: Opportunities and challenges,” *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, 2016.
- [7] Q. Chen, H. Zhu, L. Yang, X. Chen, S. Pollin, and E. Vinogradov, “Edge computing assisted autonomous flight for UAV: Synergies between vision and communications,” *IEEE Commun. Mag.*, vol. 59, no. 1, pp. 28–33, 2021.
- [8] 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, 2020.
- [9] C. Sun, W. Ni, and X. Wang, “Joint computation offloading and trajectory planning for UAV-assisted edge computing,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5343–5358, 2021.
- [10] H. Yu, S. Leng, and F. Wu, “Joint cooperative computation offloading and trajectory optimization in heterogeneous UAV-swarm-enabled aerial edge computing networks,” *IEEE Internet Things J.*, vol. 11, no. 10, pp. 17700–17711, 2024.
- [11] J. Zhang, L. Zhou, Q. Tang, E. C.-H. Ngai, X. Hu, and H. Zhao, “Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3688–3699, 2019.
- [12] Y. Jia, C. Zhang, Y. Huang, and W. Zhang, “Lyapunov optimization based mobile edge computing for Internet of Vehicles systems,” *IEEE Trans. Commun.*, vol. 70, no. 11, pp. 7418–7433, 2022.
- [13] M. Hevesli, A. M. Seid, A. Erbad, and M. Abdallah, “Multi-agent DRL for queue-aware task offloading in hierarchical MEC-enabled air-ground networks,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 12, no. 1, pp. 217–236, 2025.
- [14] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.
- [15] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” in *International Conference on Learning Representations (ICLR)*, 2016.
- [16] S. Fujimoto, H. van Hoof, and D. Meger, “Addressing function approximation error in actor-critic methods,” in *International Conference on Machine Learning (ICML)*, 2018.
- [17] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *International Conference on Learning Representations (ICLR)*, 2018.
- [18] X. Shen *et al.*, “UAV-driven task offloading and wireless power transfer: A fusion of Lyapunov optimization and reinforcement learning in edge computing,” *Physical Communication*, vol. 71, 2025, Art. no. 102719.
- [19] H. Yuan, M. Wang, J. Bi, S. Shi, J. Yang, and J. Zhang, “Cost-efficient task offloading in mobile edge computing with layered unmanned aerial vehicles,” *IEEE Internet Things J.*, vol. 11, no. 19, pp. 30496–30509, 2024.