

# MAESTRO-X: Distributed Orchestration of Rotary-Wing UAV Relay Swarms

Bharath Keshavamurthy\*, Matthew A. Bliss†, and Nicolò Michelusi\*

## Abstract

This work details the design of a scalable framework to orchestrate a swarm of rotary-wing UAVs serving as cellular relays to facilitate beyond line-of-sight connectivity and traffic offloading for ground users. First, a Multiscale Adaptive Energy-conscious Scheduling and TRajecotry Optimization (MAESTRO) framework is developed under single-agent specializations, whose goal is to minimize time-average service latencies for user requests under Poisson arrivals, subject to an average UAV power constraint. Formulated as a semi-Markov decision process and equipped with rate adaptation to exploit air-to-ground channel characteristics, the underlying objective imbues a multiscale structure to the solution: outer actions on radial wait velocities and terminal service positions optimize the long-term delay-power trade-off; consequently, inner actions on angular wait velocities and service trajectories minimize the instantaneous delay-power cost. A novel hierarchical competitive swarm optimization scheme is developed for UAV path planning, wherein iterative pair-wise updates devise trajectories of increasingly higher resolution. Next, this single-agent construction is eXtended to UAV swarms (MAESTRO-X) under a policy replication heuristic enabled by a decentralized command-and-control network; additional augmentations include a spread maximization scheme to suitably position idle UAVs in the swarm for potential new relay requests and a consensus-driven conflict resolution algorithm that orchestrates scheduling decisions based on delay-power costs that incorporate M/G/ $x$  queuing dynamics. Numerical evaluations show that MAESTRO-X delivers significant performance improvements vis-à-vis average service latencies and average UAV power consumption: for a swarm of 3 UAVs, MAESTRO-X delivers data payloads 12.5× faster than static UAV deployments and 3.25× faster than deep-Q networks; remarkably, a single relay optimized via MAESTRO outclasses 3 relays under joint successive convex approximation by 79%. Finally, the implementation feasibility of MAESTRO-X is validated through emulations involving the radio and vehicle control software of the NSF AERPAW platform.

## Index Terms

UAV Relays, SMDPs, Hierarchical CSO, M/G/ $x$  Queue Management

Part of this work has been supported by NSF under grants CNS-1642982 and CNS-2129015.

\*Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ.

†Electrical and Computer Engineering, Purdue University, West Lafayette, IN.

A part of this research has been submitted to Asilomar 2022 [1].

The source code for this project is available on GitHub [2].

## I. INTRODUCTION

With sustained device proliferation, enterprises across sectors have stepped-up their adoption of Unmanned Aerial Vehicles (UAVs) to gather data, survey infrastructure, monitor operations, and automate logistics [3], [4]. Inevitably, this has fostered varied academic research and industrial R&D on UAV-augmented beyond line-of-sight connectivity and traffic offloading in cellular networks: the coverage and service capabilities of an extant terrestrial radio access network can be enhanced by the mobility and maneuverability of these autonomous aerial relays [5], [6]. These assets of UAV-aided wireless networks are especially useful in military environments, disaster-relief applications, and precision agriculture. Our prior work on cognitive radio networks [7] involved performance analyses and mobility studies of an intelligent spectrum sensing and access framework applied to emulated UAV-enhanced troop deployment scenarios. When the existing wireless infrastructure was rendered inaccessible during the Big Hollow fire in WA, Verizon’s fleet of LTE-enabled drones allowed for expedient deployments of robust air-to-ground links to relay real-time imagery to an operations base several miles away [8]. In precision agriculture, UAV relays serve as additive companions to MooCall livestock monitors, AgOpenGPS seeding workflows, Ecorobotix weeding platforms, and VIGOR soil moisture sensors [9].

Unsurprisingly, the pervasive potential of such wireless networks brings along a plethora of challenges in real-world deployments [6]: on-board energy constraints of aerial platforms impacting mission times, stringent Quality-of-Service (QoS) requirements for accessible and reliable connectivity, channel stochastics of air-to-ground links in highly-mobile settings, and computational feasibility challenges in trajectory design brought on by the inherently large state and action spaces. Ergo, several works in the current literature have tried to tackle these challenges via optimization, machine learning, and reinforcement learning. Yet, various problems remain unsolved and various challenges are left unaddressed: failure to capture the uncertain system mechanics introduced by dynamic traffic generation [10]–[14], restrictions on UAV path and velocity characteristics [11], [15], inefficient centralized frameworks [16]–[18], computationally intractable joint multi-agent formulations [19]–[22], infeasible trajectory design schemes that are rendered prohibitive when scaled to large UAV swarms [10], [23], [24], and failure to account for the influence of link layer behaviors on the QoS performance of the network.

In this paper, considering these limitations in state-of-the-art approaches, we study distributed deployments of multiple power-constrained UAVs, equipped with transceiver chains, supplementing a terrestrial base station by relaying dynamically-generated data traffic from miscellaneous

sets of ground users. Incorporating computationally feasible trajectory design, rate adaptation to leverage Air-to-Ground (A2G) propagation conditions, management of queuing dynamics, and consensus-driven decentralized operations, the proposal detailed in our work constitute a scalable framework to efficiently orchestrate a swarm of autonomous aerial relays.

Constructing a realistic channel model that accounts for A2G channel stochastics in highly-mobile settings, we devise a rate adaptation approach that maximizes system throughput by leveraging large-scale path-loss conditions and small-scale fading statistics. To begin with, we constrain our study to a single UAV relay by proposing the Multiscale Adaptive Energy-conscious Scheduling and TRajecotry Optimization (MAESTRO) framework: with the underlying objective of minimizing the time-average communication service delay subject to an average UAV mobility power constraint, we formulate the problem as a Semi-Markov Decision Process (SMDP). Considering the UAV’s operations in both its waiting and communication states, this formulation and its resultant solution exhibit a multiscale structure, i.e., minimizing the long-term delay-power costs yields outer decisions on radial wait velocities and service positions via value iteration; next, given these outer decisions, greedily minimizing the instantaneous delay-power costs yields inner actions on angular wait velocities and service trajectories. A novel hierarchical competitive swarm optimization scheme is developed for UAV path planning, wherein iterative pair-wise cost-based comparisons devise trajectories of increasingly higher resolution.

We then extend the single-agent MAESTRO policy to UAV swarms (MAESTRO-X) by embedding it with suitable heuristics: coupled with multi-agent coordination over a decentralized command-and-control network, a spread maximization algorithm is associated with the optimal wait actions to efficiently position and prime the UAVs for potential new service requests; a consensus-driven conflict resolution algorithm is designed to orchestrate scheduling decisions based on the delay-power costs that incorporate M/G/ $x$  queuing dynamics, with the goal of choosing the best possible UAV relay for the active user request. Finally, we demonstrate the implementation viability of MAESTRO-X via emulations on the NSF AERPAW platform.

A condensed contrast between our approach and those in the relevant literature is presented in Table I. Analogous to the columns of Table I, we discuss this differentiation in detail next.

**Related Work:** First, reviewing single-relay formulations in existing works, we observe non-adaptive schemes [10]–[14] designed for applications where the ground users possess local storage or aggregation capabilities allowing for deterministic arrivals of data packets. Yet, practical deployments involve dynamically-generated traffic from heterogeneous users, each with

Paper	Adaptive Control	Channel Model	UAV Trajectory Design		Deployment of UAVs	Multi-UAV Construction	Overall Formulation	Link Layer Model	
			Mobility	Velocity				Schedule	Queue
This	Yes	A2G	Dynamic	Variable	Distributed	Decoupled	Model-based	Yes	Yes
[10]	No	FSPL	Dynamic	Variable	Single	-	Model-based	Yes	No
[16]	No	A2G	Dynamic	Variable	Centralized	Joint	Model-based	Yes	No
[19]	No	A2G	Restricted	Fixed	Distributed	Joint	Model-free	No	No
[11]	No	FSPL	Dynamic	Fixed	Single	-	Model-based	Yes	No
[12]	Yes	FSPL	Dynamic	Variable	Single	-	Model-based	Yes	No
[20]	No	FSPL	Restricted	Fixed	Distributed	Joint	Model-free	Yes	No
[13]	No	A2G	Static	-	Single	-	Model-based	No	No
[17]	No	A2G	Static	-	Distributed	Joint	Model-based	Yes	No
[18]	Yes	FSPL	Static	-	Distributed	Joint	Model-based	No	No
[25]	No	FSPL	Dynamic	Fixed	Centralized	Joint	Model-based	Yes	No
[26]	No	A2G	Static	-	Centralized	Joint	Model-based	No	No
[27]	No	A2G	Restricted	Fixed	Distributed	Decoupled	Model-free	No	No
[21]	Yes	FSPL	Static	-	Distributed	Joint	Model-free	No	Yes
[22]	No	A2G	Static	-	Distributed	Joint	Model-free	No	No
[14]	No	A2G	Dynamic	Variable	Single	-	Model-based	Yes	No

TABLE I: A comparison of the features of our framework with those of relevant schemes in the literature. varying degrees of QoS mandates and technological prowess [3], [4]. In contrast to the non-adaptive literature, we consider dynamic traffic generation from random user deployments, thereby constructing a control framework that is adaptive to uncertain system dynamics.

A common approach to solve for optimal service schedules and their associated trajectories has been Successive Convex Approximation (SCA) [10]–[12], [14]. Apart from being computationally intractable to accommodate dynamic data traffic due to their prohibitively large convergence times, SCA approaches rely on first-order Taylor approximations of the objective and the constraints in their optimization problem to enforce convexity, which introduces inaccuracies into the model. Also, these works employ Free Space Path-Loss (FSPL) communication models that fail to account for the A2G channel characteristics inherent in UAV-assisted wireless networks, and preclude the adoption of rate adaptation to leverage small- and large-scale A2G conditions. In our work, in addition to accurately modeling A2G channel conditions and employing a computationally efficient rate adaptation scheme at all the transmitters to efficiently exploit said conditions to maximize throughput, there are no such approximations in our formulation.

For trajectory design, we propose a Competitive Swarm Optimization (CSO) [24] approach to bypass the computational infeasibility encountered by SCA-oriented designs. Unlike SCA, CSO does not depend on the specific problem structure to work effectively and can thus accommodate realistic A2G propagation conditions and rate adaptation, as we do in our paper. Contrary to the limited update scope of Particle Swarm Optimization (PSO) [23], in which particle updates are driven by the individual and the swarm, CSO exhibits superior performance on several benchmarks [24] since it involves a more efficient update strategy wherein pair-wise competition is invoked between particles, permitting the winners to advance to the next iteration and the

loser particles to learn from the winners. PSO has been used in [28], [29] to optimize static hovering positions, and in [11], [15] to optimize trajectories with restrictions on path structures (e.g., moving along the circumference of a circle [15]) or velocities (e.g., fixed [11]). Moreover, in order to scale CSO to higher dimensional trajectory optimization problems, we embed CSO within a Hierarchical wrapper (HCSO), which enables the design of optimal series of way-points and velocities of increasingly higher resolution, with no such path structures imposed.

Next, shifting our attention to swarm orchestration frameworks, we find inefficient solutions such as centralized deployments [16], [25], [26] in which an aggregation gateway coordinates the operations of the UAV relays; or either joint multi-relay optimization methods [16]–[18] or model-free formulations consisting of combined state and action spaces [19]–[22]. Centralized swarm deployments bring in the need for additional capital and operational expenditure; and joint multi-UAV constructions lead to prohibitively large solution spaces resulting in prohibitive convergence times. Mindful of such considerations, we present an orchestration framework suitable for distributed deployments of two or more UAVs in a swarm by coupling our single-agent policy with spread maximization and consensus-driven link-layer prescient conflict resolution over a common control channel, and replicating it across the swarm. This approach eliminates the need for a centralized aggregation center, mitigates the computational overhead and infeasibility encountered by joint multi-relay models, and facilitates a seamless incorporation of the  $M/G/x$  queuing dynamics into our communication scheduling analyses. Additionally, although model-free control schemes in [19]–[22], [27] consider unknown system dynamics in their formulations to solve for the optimal scheduling and trajectory design solution, they fail to effectively capture the problem structure, typically resulting in slow policy convergence times; in contrast, we use a model-based approach, by casting the problem as an SMDP which effectively captures the temporal irregularity seen in the state transitions of UAV-aided non-terrestrial networks.

To the best of our knowledge, no other work in the state-of-the-art develops a scalable orchestration framework for UAV relay swarms while incorporating practical features such as dynamic data traffic from miscellaneous sets of randomly distributed ground users; a feasible trajectory design solution, without unrealistic assumptions on UAV mobility; energy-conscious control of the UAV relays during periods of idle operation; efficient exploitation of A2G channel characteristics via rate adaptation, without unnecessary approximations in the primal problem; inclusion of  $M/G/x$  queuing mechanics in the scheduling decisions; and decoupled executions of a single trained policy over realistic consensus-driven decentralized swarm deployments.

The rest of the paper is organized as follows: Sec. II introduces the system model; Sec. III elucidates the design of MAESTRO, our single-agent SMDP formulation; Sec. IV describes the algorithms inherent in our policy optimization process; Sec. V outlines our multi-agent enhancements (MAESTRO-X); Sec. VI chronicles our numerical evaluations along with our emulations on the NSF AERPAW platform; and finally, Sec. VII lists our conclusions.

## II. SYSTEM MODEL

Consider the deployment scenario illustrated in Fig. 1a, wherein a swarm of  $N_U$  rotary-wing Unmanned Aerial Vehicles (UAVs)—each equipped with an on-board transceiver chain—operate as cellular relays to supplement the coverage and service capabilities of a terrestrial Base Station (BS) by relaying data traffic dynamically-generated by users at ground level, i.e., Ground Nodes (GNs). The BS is located at the center of the circular cell (of radius  $a$ ), at height  $H_B$ , while the UAVs operate at a fixed height of  $H_U$ . The GNs are distributed uniformly at random throughout the cell, with a density of  $\lambda_G$  [GNs per unit area]. The BS utilizes  $k$  orthogonal channels to serve the GNs simultaneously via an Orthogonal Frequency Division Multiple Access (OFDMA) strategy; on the other hand, the UAV relays are restricted to serve one GN at a time through a decode-and-forward scheme. All channels are assumed to have a bandwidth of  $B$ . Next, we set up our communication model focusing on the uplink only<sup>1</sup>, i.e., traffic requests generated by the GNs are transmitted to the BS, either directly or using a UAV in the swarm as a relay.

**Communication Model:** Each GN generates uplink transmission requests of  $L$  bits, according to a Poisson process with rate  $\lambda_{R|G}$  [requests per GN per unit time]. Coupled with the random deployment of GNs, uplink requests arrive in time according to a Poisson process with rate  $\lambda_R = \lambda_G \cdot \lambda_{R|G}$  [requests per unit time per unit area]. Thus,  $\Lambda \triangleq \lambda_R \pi a^2$  [requests per unit time] is the overall request arrival rate over the circular cell. Since a new request is uniformly distributed in the cell area, the position  $(r, \theta)$  of the GN originating the request—expressed in polar coordinates with respect to the BS—has angular coordinate  $\theta$  uniform in  $[0, 2\pi]$ , and radial coordinate with probability density function  $f_R(r) = \frac{2r}{a^2} \mathbb{I}(r \leq a)$ , where  $\mathbb{I}(\cdot)$  represents the indicator function.

A fully-connected mesh network is overlaid on the BS and the UAVs in the cell in order to establish a command-and-control network, allocating the band-edges of the spectrum under use as control channels. Since the packets exchanged among the mesh nodes over the control channel constitute short frames relative to the large data payloads generated by the GNs (and

<sup>1</sup>This can be extended to both uplink and downlink traffic by creating a state variable differentiating between the two.

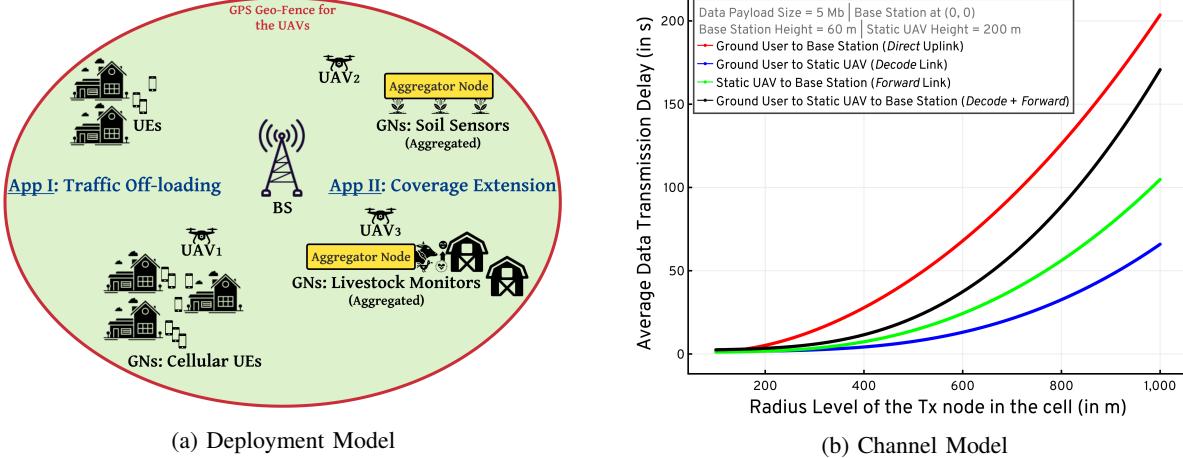


Fig. 1: (a) A terrestrial BS aided by 3 UAVs serving as cellular relays for a diverse set of GNs: traffic offloading for cellular UEs, and coverage extensions for livestock monitors and soil sensors; (b) The service latencies experienced by GNs for 5 Mb data payloads under both direct-BS as well as Decode-and-Forward strategies.

communicated over orthogonal data channels), it is reasonable to neglect the latencies involved in these control operations. When a GN decides to upload its data, it informs the BS—over the control channel—about the need for an uplink transmission of  $L$  bits, and includes its physical location in this preliminary request for service. Considering potential delay-power costs for this request, the BS and the UAVs coordinate over the control network to arrive at a consensus on the best scheduling decision: if direct transmission is chosen, the BS assigns a data channel  $k \in \{1, 2, \dots, N_B\}$  to the GN and instructs it to begin a direct transmission to the BS (without involving the use of a UAV relay); else, if relaying the data payload through UAV  $i$  is determined to be the most efficient choice, such UAV instructs the GN to begin transmission over its designated pre-determined data channel  $k_{U_i} \in \{1, 2, \dots, N_U\}$ . A Decode-and-Forward (D&F) strategy underlies the communication process encountered in the latter case: while moving along a designed trajectory (a sequence of way-points and velocities), UAV  $i$  first receives the entire data payload from the GN over channel  $k_{U_i}$  (decode) and subsequently transmits it to the BS over the same channel (forward). As a result of these scheduling decisions, the GN→BS, GN→UAV, and UAV→BS links must be studied. We discuss the channel underlying these links next.

**A2G Channel Model:** For a generic link, we denote the flat-fading channel coefficient as  $h \triangleq \sqrt{\beta}g$ , where  $\beta$  captures the large-scale channel variations, and  $g$  with  $\mathbb{E}[|g|^2] = 1$  is the small-scale fading component. We model the large-scale component as  $\beta = \beta_{\text{LoS}}(d) \triangleq \beta_0 d^{-\alpha}$  for line-of-sight (LoS) and  $\beta = \beta_{\text{NLoS}}(d) \triangleq \kappa \beta_0 d^{-\tilde{\alpha}}$  for non-LoS (NLoS) links, where  $\beta_0$  is the pathloss referenced at a distance of 1 meter,  $2 \leq \alpha \leq \tilde{\alpha}$  are the LoS and NLoS path-loss exponents,  $\kappa \in (0, 1]$  captures the additional NLoS attenuation, and  $d$  denotes the Tx-Rx Euclidean distance [10].

We model the LoS and NLoS probabilities as functions of the elevation angle  $\varphi \in (0^\circ, 90^\circ]$ , i.e.,

$$P_{\text{LoS}}(\varphi) = \frac{1}{1 + z_1 \exp(-z_2 [\varphi - z_1])}, \quad P_{\text{NLoS}}(\varphi) = 1 - P_{\text{LoS}}(\varphi), \quad (1)$$

where  $z_1$  and  $z_2$  are parameters specific to the propagation environment (e.g., rural, suburban, and urban) [30]. The distribution of the small-scale fading component  $g$  also depends on the LoS or NLoS link state [31]. Specifically, for a LoS link, as in [14], we model  $g$  as Rician fading with a  $\varphi$ -dependent  $K$ -factor  $K(\varphi) = k_1 \exp(k_2 \varphi)$ , where coefficients  $k_1$  and  $k_2$  are determined by the propagation environment [14]. For a NLoS link, we model  $g$  as Rayleigh fading [31] (Rician with  $K=0$ ). Given  $h$ , the link capacity is  $C(h) = B \cdot \log_2 \left( 1 + \frac{|h|^2 P}{N_0 B \Gamma} \right)$ , where  $P$  is the transmission power,  $N_0$  is the noise power spectral density at the receiver,  $B$  is the channel bandwidth, and  $\Gamma$  is the Signal-to-Noise Ratio (SNR) gap between practical modulation-and-coding schemes and theoretical Gaussian signaling [14]. We assume that other sources of signal degradation, such as the Doppler effect, are well-compensated at the receiver (e.g., see the approaches in [32]).

Since the large-scale components typically vary slowly relative to the rate of acquisition of Channel State Information (CSI), we assume that the current large-scale parameters  $(\beta, K)$  are known at the transmitter's side throughout the communication process, which enables rate-control at the transmitter; on the other hand, small-scale fading conditions vary at a much faster timescale, hence cannot be tracked at the transmitter, which may result in outages when the selected rate exceeds the channel capacity  $C(h)$ . Thus, given  $(\beta, K)$  and a transmission rate of  $\Upsilon$  [bits per second], we define the outage probability  $P_{\text{out}}(\Upsilon, \beta, K) \triangleq \mathbb{P}(C(\sqrt{\beta} g) < \Upsilon) | \beta, K = \mathbb{P}(|g|^2 < u(\Upsilon, \beta))$ , where  $u(\Upsilon, \beta) \triangleq N_0 B \Gamma (2^{\Upsilon/B} - 1) / (\beta P)$ . Since  $2(K+1)|g|^2$  has a non-central  $\chi^2$  distribution with 2 degrees of freedom and non-centrality parameter  $2K$ , we find that

$$P_{\text{out}}(\Upsilon, \beta, K) = 1 - Q_1 \left( \sqrt{2K}, \sqrt{2(K+1)u(\Upsilon, \beta)} \right), \quad (2)$$

where  $Q_1(\cdot, \cdot)$  is the standard Marcum  $Q$ -function [14]. Note that when  $K=0$  (Rayleigh fading for NLoS links), the function specializes to  $Q_1 \left( 0, \sqrt{2u(\Upsilon, \beta)} \right) = \exp(-u(\Upsilon, \beta))$ . We assume that the small-scale fading is averaged out across time, yielding the expected throughput as,

$$R(\Upsilon, \beta, K) = \Upsilon \cdot (1 - P_{\text{out}}(\Upsilon, \beta, K)) = \Upsilon \cdot Q_1 \left( \sqrt{2K}, \sqrt{2(K+1)u(\Upsilon, \beta)} \right). \quad (3)$$

In our model, we permit rate adaptation at the transmitter based on the large-scale parameters  $(\beta, K)$ , coordinated through the control channel via CSI feedback. The transmission rate  $\Upsilon$  is cho-

sen to maximize the expected throughput given  $(\beta, K)$ , i.e.,  $\Upsilon^*(\beta, K) \triangleq \arg \max_{\Upsilon \geq 0} R(\Upsilon, \beta, K)$ . To solve this problem, let  $Z \triangleq \sqrt{\frac{2\beta P}{N_0 B \Gamma}} u(\Upsilon, \beta)$ , so that  $\Upsilon = B \log_2 (1 + \frac{1}{2} Z^2) \triangleq f(Z)$ . It follows that  $\Upsilon^*(\beta, K) = f(Z^*(\beta, K))$ , where  $Z^*(\beta, K) \triangleq \arg \min_{Z \geq 0} h(Z)$  and  $h(Z) \triangleq -\ln f(Z) - \ln Q_1 \left( \sqrt{2K}, \sqrt{\frac{(K+1)\beta P}{N_0 B \Gamma}} Z \right)$ . Since the function  $h(Z)$  is convex,  $Z^*(\beta, K)$  can be found efficiently using a bisection method. Upon determining the optimal transmission rate  $\Upsilon^*(\beta, K)$ , we define the optimized throughput, as a function of the large-scale conditions, as  $R^*(\beta, K) \triangleq R(\Upsilon^*(\beta, K), \beta, K)$ . Further assuming that the LoS and NLoS conditions are averaged out in the temporal and spatial dimensions, the average link throughput with rate adaptation is

$$\bar{R}(d, \varphi) \triangleq P_{\text{LoS}}(\varphi) \cdot R^*(\beta_{\text{LoS}}(d), K(\varphi)) + P_{\text{NLoS}}(\varphi) \cdot R^*(\beta_{\text{NLoS}}(d), 0). \quad (4)$$

This throughput is then specialized to the three distinct communication links by expressing the transmission powers, the environment-specific parameters ( $z_1, z_2, k_1, k_2$ ), the large-scale parameters  $(\beta, K)$ , and the LoS or NLoS probabilities (1) based on the spatial configuration, i.e.,  $d$  and  $\varphi$ . Specifically, for the GN→BS link, we let  $\bar{R}_{GB}(r)$  be the throughput with the GN in position  $(r, \theta)$ , computed by setting the GN-BS distance as  $d = \sqrt{H_B^2 + r^2}$  and the elevation angle as  $\varphi = \sin^{-1}(H_B/d)$  in (4). Similarly, for the GN→UAV link, we let  $\bar{R}_{GU}(r_{GU})$  be the throughput when the GN-UAV distance (projected onto the  $x-y$  plane) is  $r_{GU}$ , computed by setting the GN-UAV Euclidean distance as  $d = \sqrt{r_{GU}^2 + H_U^2}$  and the elevation angle as  $\varphi = \sin^{-1}(H_U/d)$  in (4). Finally, for the UAV→BS link, we let  $\bar{R}_{UB}(r_{UB})$  be the throughput when the  $x-y$  projected UAV-BS distance is  $r_{UB}$ , computed by setting the GN-UAV Euclidean distance as  $d = \sqrt{r_{UB}^2 + (H_U - H_B)^2}$  and the elevation angle as  $\varphi = \sin^{-1}((H_U - H_B)/d)$  in (4). As depicted in Fig. 1b, in a system with the rate adaptation capabilities described above, the poor QoS performance experienced by GNs farther away from the BS—due to the deterioration in LoS probabilities with distance—motivates the need for aerial relays to ensure QoS guarantees.

**UAV Mobility Power Model:** For a rotary-wing UAV, since its communication power needs ( $\approx 5$  W) are dwarfed by its mobility power requirements ( $\approx 1000$  W), we model the on-board energy constraints of the UAV purely in terms of its flight power. From [10], we have that

$$P_{\text{mob}}(V) = P_1 \left( 1 + \frac{3V^2}{U_{\text{tip}}^2} \right) + P_2 \left( \sqrt{1 + \frac{V^4}{4v_0^4}} - \frac{V^2}{2v_0^2} \right)^{1/2} + P_3 V^3, \quad (5)$$

where  $V$  is the horizontal flying velocity,  $P_j, j \in \{1, 2, 3\}$  are scaling constants,  $U_{\text{tip}}$  is the rotor blade tip velocity, and  $v_0$  is the mean rotor induced velocity while hovering. From [10], hovering

requires 1370 W, whereas flying at the power-minimizing velocity of  $22 \text{ m s}^{-1}$  only consumes 940 W. This suggests that, as we will show, the mobility of the UAVs can be exploited to reduce their power consumption, while simultaneously ensuring that the QoS mandates for GNs in the cell are successfully met. Hence, our objective is to solve for an energy-conscious adaptive service scheduling and trajectory optimization scheme wherein a distributed swarm of UAV relays are employed to minimize the time-average communication service delay experienced by GNs in the cell, randomly generating uplink transmission requests, subject to an average per-UAV mobility power constraint. To this end, in the next section, we formulate a framework specialized to single UAV deployments, which is then extended to UAV swarms in Sec. V.

### III. MAESTRO: A SEMI-MARKOV DECISION PROCESS FORMULATION

In this section, we specialize the deployment, communication, and channel models detailed in Sec. II to single UAV relay settings. Accordingly, we describe the mathematical constructions involved in the design of our solution framework to minimize the time-average service delay experienced by the GNs in the cell, subject to an average UAV mobility power constraint via a Semi-Markov Decision Process (SMDP) formulation. The effective traffic rate experienced by a single UAV is  $\Lambda' \triangleq \frac{\Lambda}{N_U}$  [requests per unit time per UAV], which is assumed in this section in place of the overall rate  $\Lambda$ . Let  $\mathbf{q}_U(t) = (r_U(t), \theta_U(t))$  be the polar coordinate of the UAV at time  $t$ , projected onto the  $x-y$  plane, where  $r_U(t) \in \mathbb{R}_+$  and  $\theta_U(t) \in [0, 2\pi)$  denote the UAV's radius and angle with respect to the BS (cell center). This specialized setup is illustrated in Fig. 2.

We note that the operations of the UAV relay can be split into the following phases. In the waiting phase, no GN requests are being served by the UAV, which thus moves according to a waiting policy, until a new request is received. When a new GN request is received, say from position  $(r, \theta)$ , the system transitions to the request scheduling phase, where it is determined whether the GN should transmit its data payload directly to the BS, or relay it through the UAV. If direct transmission is selected, the system immediately re-enters the waiting phase, as the UAV remains free to serve other requests; else, the system enters the UAV relay phase, in which the GN relays its data payload through the UAV using the D&F protocol; upon the completion of this relay service, the system re-enters the waiting phase. Note that the BS can accommodate simultaneous transmissions using orthogonal channels (see Sec. II): therefore, new requests received during the UAV relay phase are directly served by the BS. In this section, we assume that there are always channels available, so that these requests can be immediately served by the BS; the effect of the limited number of channels is studied in Sec. V via queuing models.

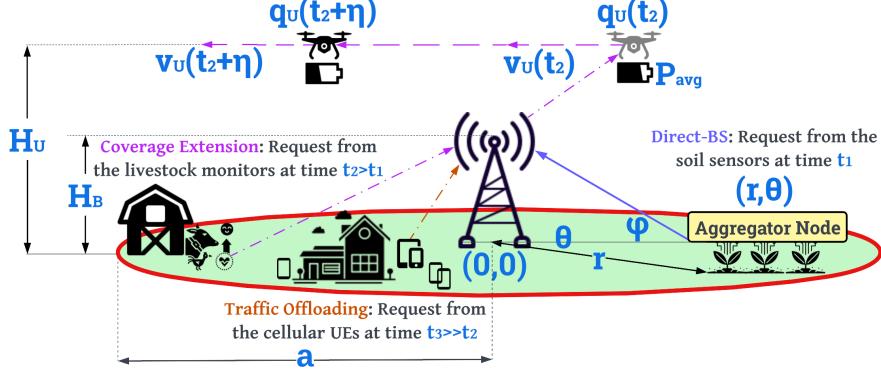


Fig. 2: The single-agent specialization of our generalized deployment depicted in Fig. 1a.

Next, we formulate the average communication delay and UAV energy consumption under a given policy  $\mu$  that defines the request scheduling, communication strategy, and UAV trajectory (formally defined later). We define a decision interval as the time duration spanning the start of a waiting phase, the subsequent request scheduling phase when a GN request is received, until the system re-enters the waiting phase after scheduling a direct transmission to the BS, or following the UAV relay phase. Consider the  $u$ th such decision interval; let  $\Delta_u^{(w)}$  be the time to wait for a new request, and  $\Delta_u^{(s)}$  be the time to serve it, either through the BS (denoted with the scheduling decision  $\xi_u=0$ ) or through the UAV ( $\xi_u=1$ ). Then, the  $u$ th decision interval duration is  $\Delta_u=\Delta_u^{(w)}+\xi_u\Delta_u^{(s)}$ , since the UAV enters the waiting phase immediately if a direct transmission to the BS is scheduled. Let  $N_u \geq 0$  be the number of additional requests received during the UAV relay phase of the  $u$ th decision period; since these additional requests are served directly by the BS, let  $\Delta_{u,i}^{(bs)}$ ,  $i = \{1, \dots, N_u\}$  be their communication delays. Let  $E_u$  be the UAV mobility energy expended during the  $u$ th decision interval. Let  $M_t$  be the total number of decision intervals completed up to time  $t$ . We define  $\bar{D}_\mu$ , the expected average communication delay per GN request under  $\mu$ , as the total communication delay accrued until time  $t$ , over the total number of requests served until time  $t$ ; similarly, we define  $\bar{P}_\mu$ , the expected average UAV power, as the total energy consumption over the total duration of the  $M_t$  intervals up to time  $t$ , i.e.,

$$\bar{D}_\mu \triangleq \lim_{t \rightarrow \infty} \mathbb{E}_\mu \left[ \frac{\frac{1}{M_t} \sum_{u=1}^{M_t} (\Delta_u^{(s)} + \xi_u \sum_{i=1}^{N_u} \Delta_{u,i}^{(bs)})}{\frac{1}{M_t} \sum_{u=1}^{M_t} (1 + \xi_u N_u)} \right], \quad \bar{P}_\mu \triangleq \lim_{t \rightarrow \infty} \mathbb{E}_\mu \left[ \frac{\frac{1}{M_t} \sum_{u=1}^{M_t} E_u}{\frac{1}{M_t} \sum_{u=1}^{M_t} \Delta_u} \right]. \quad (6)$$

Note that  $\bar{D}_\mu$  in (6) captures the delays of all requests, i.e., those relayed through the UAV ( $\xi_u=1$ ), those transmitted directly to the BS ( $\xi_u=0$ ), as well as the  $N_u$  additional requests served directly by the BS during the UAV relay phase. The goal is to solve  $\bar{D}_\mu^* = \min_\mu \bar{D}_\mu$  s.t.  $\bar{P}_\mu \leq P_{avg}$ , with optimal policy  $\mu^*$ . To simplify these expressions, let  $\bar{\mathbb{E}}_\mu[C_u] \triangleq \lim_{t \rightarrow \infty} \mathbb{E}_\mu[\frac{1}{M_t} \sum_{u=1}^{M_t} C_u]$  be

a shorthand notation to denote the long-term average cost  $C_u$  per decision interval, and let  $\bar{E}_\mu \triangleq \bar{\mathbb{E}}_\mu [E_u]$ ,  $\bar{T}_\mu \triangleq \bar{\mathbb{E}}_\mu [\Delta_u]$ ,  $\bar{N}_\mu \triangleq \bar{\mathbb{E}}_\mu [1 + \xi_u N_u]$ ,  $\bar{W}_\mu^{(s)} \triangleq \bar{\mathbb{E}}_\mu [\Delta_u^{(s)}]$ ,  $\bar{W}_\mu^{(bs)} \triangleq \bar{\mathbb{E}}_\mu [\xi_u \sum_{i=1}^{N_u} \Delta_{u,i}^{(bs)}]$  be the expected average UAV energy expenditure ( $\bar{E}_\mu$ ), duration ( $\bar{T}_\mu$ ), number of requests served ( $\bar{N}_\mu$ ), delay of requests for which a scheduling decision is made ( $\bar{W}_\mu^{(s)}$ ), and total delays of requests served directly by the BS during the UAV relay phase ( $\bar{W}_\mu^{(bs)}$ ), per decision interval. Since  $\bar{P}_\mu = \bar{E}_\mu / \bar{T}_\mu$  and  $\bar{D}_\mu = (\bar{W}_\mu^{(s)} + \bar{W}_\mu^{(bs)}) / \bar{N}_\mu$  from Little's Law [33], we can then reformulate

$$\bar{D}_\mu^* = \min_{\mu} \frac{\bar{W}_\mu^{(s)} + \bar{W}_\mu^{(bs)}}{\bar{N}_\mu} \text{ s.t. } \bar{E}_\mu - P_{\text{avg}} \bar{T}_\mu \leq 0. \quad (7)$$

Let  $\mu_{BS}$  denote the policy wherein all requests are served by the BS and the UAV flies around at the power-minimizing speed  $V_{\min}$ . Clearly, this policy is feasible for the above problem, since the UAV uses minimum flying power. Since the delay to serve a request from a GN in position  $(r, \theta)$  by direct transmission to the BS is  $L/\bar{R}_{GB}(r)$ , the expected delay under policy  $\mu_{BS}$  is obtained by computing the expectation with respect to the radial coordinate,  $\bar{D}_{BS} \triangleq \int_0^a \frac{L}{\bar{R}_{GB}(r)} f_R(r) dr$ . Clearly, by optimizing over  $\mu$ ,  $\bar{D}_\mu^* \leq \bar{D}_{BS}$ : this suggests that optimization of the UAV trajectory and scheduling decisions is key to improving the delay performance to serve requests. Note the inherent complexity that is required to solve the optimization problem (7): as the policy varies, the delay metric changes both the numerator and denominator terms of the objective function, which cannot be solved with standard dynamic programming algorithms. To address this challenge, we now propose an alternative optimization criterion based on upper- and lower-bounds to  $\bar{D}_\mu$ .

Let  $\bar{W}_\mu \triangleq \bar{W}_\mu^{(s)} + \bar{W}_\mu^{(bs)}$ . If  $\xi_u = 1$ , then additional requests received during the UAV relay phase are served directly by the BS, with delay  $L/\bar{R}_{GB}(r)$  for a GN in position  $(r, \theta)$ . Therefore, the expected average communication delay to serve these additional requests is  $\mathbb{E}[\Delta_{u,i}^{(bs)}] = \bar{D}_{BS}$ , yielding  $\bar{W}_\mu = \bar{W}_\mu^{(s)} + \bar{D}_{BS}(\bar{N}_\mu - 1)$  and  $\bar{D}_\mu = \frac{\bar{W}_\mu}{\bar{N}_\mu} = \frac{\bar{W}_\mu^{(s)}}{\bar{N}_\mu} + \left(1 - \frac{1}{\bar{N}_\mu}\right) \bar{D}_{BS}$ . Let  $\mu$  be any policy (including the optimal one) that satisfies  $\bar{D}_\mu \leq \bar{D}_{BS}$ : under such policy, since  $\bar{N}_\mu \geq 1$ , the expression above implies that  $\bar{W}_\mu^{(s)} \leq \bar{D}_\mu \leq \bar{D}_{BS}$ . Moreover, since  $\mathbb{E}[N_u | \Delta_u^{(s)}] = \Delta_u^{(s)} \Lambda'$  and  $\xi_u \leq 1$ , it follows that  $\bar{N}_\mu \leq 1 + \Lambda' \bar{W}_\mu^{(s)}$  with equality if the UAV always serves requests. This implies that  $\bar{W}_\mu^{(s)} \leq \bar{D}_\mu \leq \bar{W}_\mu^{(s)} \frac{1 + \Lambda' \bar{D}_{BS}}{1 + \Lambda' \bar{W}_\mu^{(s)}} \leq \bar{D}_{BS}$ . Observing that both the lower and upper bounds of  $\bar{D}_\mu$  are increasing functions of  $\bar{W}_\mu^{(s)}$ , we replace (7) with the surrogate  $\min_{\mu} \bar{W}_\mu^{(s)}$  s.t.  $\bar{E}_\mu - P_{\text{avg}} \bar{T}_\mu \leq 0$ , which will be the focus of our subsequent analyses. To solve this, we employ the Lagrangian

$$g(\nu) = \min_{\mu} (\bar{W}_\mu^{(s)} + \nu(\bar{E}_\mu - P_{\text{avg}} \bar{T}_\mu)) = \min_{\mu} \lim_{t \rightarrow \infty} \mathbb{E}_\mu \left[ \frac{1}{M_t} \sum_{u=1}^{M_t} (\Delta_u^{(s)} + \nu(E_u - P_{\text{avg}} \Delta_u)) \right], \quad (8)$$

where  $\nu$  is the dual variable, optimized by solving  $\max_{\nu \geq 0} g(\nu)$ . We now demonstrate that for a given  $\nu \geq 0$ , (8) can be cast as a Semi-Markov Decision Process (SMDP) and solved with dynamic programming tools: next, we discuss its states, actions, transitions, and policy.

**States:** The state is defined by the UAV position  $\mathbf{q}_U$ , an element of the set  $\mathcal{Q}_{\text{UAV}} \triangleq \mathbb{R}_+ \times [0, 2\pi)$  (polar coordinates), and the position of the GN originating traffic  $\mathbf{q}_G$ , taking values from the set  $\mathcal{Q}_{\text{GN}} \triangleq [0, a] \times [0, 2\pi)$ . The state space is then  $\mathcal{S} = \mathcal{S}_{\text{wait}} \cup \mathcal{S}_{\text{comm}}$ , where  $\mathcal{S}_{\text{wait}} = \mathcal{Q}_{\text{UAV}}$  is the set of waiting states and  $\mathcal{S}_{\text{comm}} = \mathcal{Q}_{\text{UAV}} \times \mathcal{Q}_{\text{GN}}$  is the set of communication states. Crucial to the definition of the SMDP is how the system is sampled in time to define Markovian dynamics in the evolution of the sampled states: accordingly, we define the actions available in each state  $s \in \mathcal{S}$  and the transition probabilities, along with the time duration  $T(s; a)$ , the UAV energy usage  $E(s; a)$ , and the request service delay  $\Delta(s; a)$  metrics accrued in state  $s$  under action  $a$ .

**Waiting states' actions, transitions, and metrics:** If the UAV is in waiting state  $s_n = \mathbf{q}_U \in \mathcal{S}_{\text{wait}}$  at time  $t$ , i.e., it is in position  $\mathbf{q}_U(t) = \mathbf{q}_U = (r_U, \theta_U)$  with no active requests, then the actions available are to move the UAV with radial and angular velocity components  $(v_r, \theta_c)$ , over an arbitrarily small duration  $\Delta_0 \ll 1/\Lambda'$ . Under a maximum velocity constraint  $V_{\text{max}}$ , this waiting-state action space is given by  $\mathcal{A}_{\text{wait}}(r_U) \triangleq \left\{ (v_r, \theta_c) \in \mathbb{R}^2 \mid \sqrt{v_r^2 + r_U^2 \cdot \theta_c^2} \leq V_{\text{max}} \right\}$ , where  $v_U = \sqrt{v_r^2 + r_U^2 \theta_c^2}$  is the velocity expressed with respect to polar coordinates. Upon choosing action  $\mathbf{a} = (v_r, \theta_c) \in \mathcal{A}_{\text{wait}}(r_U)$ , the communication delay is  $\Delta(s; a) = 0$ , since there is no ongoing communication; the duration of a waiting state visit is  $T(s; a) = \Delta_0$ , during which the UAV uses an amount of energy  $E(s; a) = \Delta_0 P_{\text{mob}}(v_U)$  to move at velocity  $v_U$ . The new state is then sampled at time  $t + \Delta_0$ , with the UAV moved to the new position  $\mathbf{q}_U(t + \Delta_0) \approx (r_U, \theta_U) + (v_r, \theta_c)\Delta_0$ . With probability  $e^{-\Lambda'\Delta_0}$ , no new request is received in the time interval  $[t, t + \Delta_0]$ , so that the new state is a waiting state. Otherwise, a new request is received from a GN in position  $(r, \theta)$ , so that the new state is a communication state. Accordingly, the transition probability model from the waiting state  $s_n = \mathbf{q}_U \in \mathcal{S}_{\text{wait}}$  under action  $\mathbf{a}_n = (v_r, \theta_c) \in \mathcal{A}_{\text{wait}}(r_U)$  is mathematically described by

$$\mathbb{P}(s_{n+1} = \mathbf{q}_U + \mathbf{a}_n \Delta_0 | s_n, \mathbf{a}_n) = e^{-\Lambda' \Delta_0}, \quad (9)$$

$$\mathbb{P}(s_{n+1} = (\mathbf{q}_U + \mathbf{a}_n \Delta_0, \mathbf{q}'_G) \text{ with } \mathbf{q}'_G \in \mathcal{F} | s_n, \mathbf{a}_n) = \frac{A(\mathcal{F})}{\pi a^2} \cdot (1 - e^{-\Lambda' \Delta_0}), \quad \forall \mathcal{F} \subseteq \mathcal{Q}_{\text{GN}},$$

where  $A(\mathcal{F})$  is the area of region  $\mathcal{F}$ , since requests are uniformly distributed in the cell.

**Communication states' actions, transitions, and metrics:** Upon reaching a communication state  $s_n = (\mathbf{q}_U, \mathbf{q}_G) \in \mathcal{S}_{\text{comm}}$  at time  $t$ , the system must serve a GN request at position  $\mathbf{q}_G = (r, \theta)$ .

The BS first determines the scheduling decision  $\xi \in \{0, 1\}$ . If  $\xi=0$ , the GN transmits directly to the BS, and the next state is sampled immediately after the decision, so that it is the waiting state  $s_{n+1} = q_U \in \mathcal{S}_{\text{wait}}$  with probability 1. We denote this action as  $a=(0, [ ])$ , with cost metrics  $\Delta(s_n; a) = \frac{L}{\bar{R}_{GB}(r)}$ ,  $E(s_n; a) = 0$ ,  $T(s_n; a) = 0$ , since direct transmissions occur at throughput  $\bar{R}_{GB}(r)$  and the system moves immediately to the waiting state  $q_U \in \mathcal{S}_{\text{wait}}$  resulting in the action duration and energy expenditure being 0. On the other hand, if  $\xi=1$ , the UAV uses the D&F protocol described next, while following a trajectory starting from its current position  $q_U$  and ending in position  $q'_U$ . We denote this action as  $a=(1, q_U \rightarrow q'_U)$ . In the first phase (of duration  $t_p$ ) of the D&F protocol, the GN transmits its payload to the UAV; in the second phase (of duration  $\Delta - t_p$ ), the UAV relays the data payload to the BS. Assuming a move-and-transmit implementation [10], the trajectory  $(q_U \rightarrow q'_U)$  and the durations ( $t_p$  and  $\Delta - t_p$ ) must satisfy the data payload constraint given by (C.1) below, i.e., the entire payload of  $L$  bits is first transmitted to the UAV with rate  $\bar{R}_{GU}(r_{GU}(t+\eta))$ , where  $r_{GU}(t+\eta)$  is the GN-UAV distance (projected onto the  $x-y$  plane) at time  $t+\eta$ ; then, the UAV transmits the payload to the BS with rate  $\bar{R}_{UB}(r_{UB}(t+\eta))$ , where  $r_{UB}(t+\eta)$  is the radial position of the UAV at time  $t+\eta$ , so that the total communication delay is  $\Delta$ . In this case, the cost metrics under action  $a = (1, q_U \rightarrow q'_U)$  are  $\Delta(s_n; a) = \Delta$ ,  $E(s_n; a) = \int_0^\Delta P_{\text{mob}}(v_U(t+\eta)) d\eta$ ,  $T(s_n; a) = \Delta$ . Upon completing D&F, the UAV enters the waiting state, i.e.,  $s_{n+1} = q'_U$  at time  $t+\Delta$ . The set of feasible UAV trajectories  $\mathcal{Q}_{q_G}(q_U \rightarrow q'_U)$  starting in  $q_U$ , terminating in  $q'_U$ , to serve a GN at  $q_G$  is

$$\mathcal{Q}_{q_G}(q_U \rightarrow q'_U) = \left\{ \mathbf{p}_U : [0, \Delta] \mapsto \mathbb{R}_+ \times [0, 2\pi) \text{ s.t.} \right. \quad (10)$$

$$\left. \int_0^{t_p} \bar{R}_{GU}(r_{GU}(t+\eta)) d\eta \geq L, \quad \int_{t_p}^\Delta \bar{R}_{UB}(r_{UB}(t+\eta)) d\eta \geq L, \right. \quad (C.1)$$

$$v_U(\eta) \leq V_{\max}, \quad \forall \eta \in [0, \Delta], \quad (C.2)$$

$$\left. \mathbf{p}_U(0) = q_U, \mathbf{p}_U(\Delta) = q'_U, \quad \exists \Delta \geq 0, \exists 0 \leq t_p \leq \Delta \right\}, \quad (C.3)$$

where C.1 reflects the data payload constraints, C.2 the maximum velocity constraint, and C.3 the trajectory constraints. Then, the action space in state  $(q_U, q_G) \in \mathcal{S}_{\text{comm}}$  when  $\xi=1$  is the set  $\mathcal{Q}_{q_G}(q_U) \triangleq \cup_{q'_U \in \mathcal{Q}_{\text{UAV}}} \mathcal{Q}_{q_G}(q_U \rightarrow q'_U)$  of feasible trajectories starting in  $q_U$  that serve the GN at  $q_G$  via the D&F protocol. The overall action space of these communication states is then  $\mathcal{A}_{\text{comm}}(q_U, q_G) \triangleq \{0, [ ]\} \cup \{\{1\} \times \mathcal{Q}_{r,\theta}(r_U, \theta_U)\}$ : the set  $\{0, [ ]\}$  corresponds to  $\xi=0$  (direct-BS), while the set  $\{\{1\} \times \mathcal{Q}_{r,\theta}(r_U, \theta_U)\}$  corresponds to  $\xi=1$  (relayed service through the UAV).

**Policy  $\mu$ :** For waiting states  $\mathbf{q}_U \in \mathcal{S}_{\text{wait}}$ , the policy selects a velocity  $(v_r, \theta_c)$  from the corresponding action space i.e.,  $\mu(\mathbf{q}_U) \in \mathcal{A}_{\text{wait}}(r_U)$ . Likewise, for communication states  $(\mathbf{q}_U, \mathbf{q}_G) \in \mathcal{S}_{\text{comm}}$ , the policy selects the scheduling decision  $\xi \in \{0, 1\}$  and if  $\xi=1$ , the trajectory followed in the D&F protocol, i.e.,  $\mu(\mathbf{q}_U, \mathbf{q}_G) \in \mathcal{A}_{\text{comm}}(\mathbf{q}_U, \mathbf{q}_G)$ . With a stationary policy  $\mu$  defined, the Lagrangian metric  $L_\mu^{(\nu)} \triangleq \bar{W}_\mu^{(s)} + \nu(\bar{E}_\mu - P_{\text{avg}}\bar{T}_\mu)$  in (8) is reformulated using Little's Law [33] as

$$L_\mu^{(\nu)} = \lim_{N \rightarrow \infty} \mathbb{E}_\mu \left[ \frac{\frac{1}{N} \sum_{n=0}^{N-1} \ell_\nu(\mathbf{s}_n; \mu(\mathbf{s}_n))}{\frac{1}{N} \sum_{n=0}^{N-1} \mathbb{I}(\mathbf{s}_n \in \mathcal{S}_{\text{comm}})} \right] = \frac{1}{\pi_{\text{comm}}} \int_{\mathcal{S}} \Pi_\mu(\mathbf{s}) \ell_\nu(\mathbf{s}; \mu(\mathbf{s})) d\mathbf{s}, \quad (11)$$

where  $\Pi_\mu(\mathbf{s})$  is the steady-state probability density function of the SMDP being in a state  $\mathbf{s}$  under policy  $\mu$ ,  $\pi_{\text{comm}} = \int_{\mathcal{S}_{\text{comm}}} \Pi_\mu(\mathbf{s}) d\mathbf{s}$  is the steady-state probability that the UAV is in the communication phase in the SMDP, and  $\ell_\nu(\mathbf{s}; \mathbf{a}) \triangleq \Delta(\mathbf{s}; \mathbf{a}) + \nu(E(\mathbf{s}; \mathbf{a}) - P_{\text{avg}}T(\mathbf{s}; \mathbf{a}))$  is the overall Lagrangian metric in state  $\mathbf{s}$  under action  $\mathbf{a}$ . In (11), note that  $\sum_{n=0}^{N-1} \ell_\nu(\mathbf{s}_n; \mu(\mathbf{s}_n))$  is the total Lagrangian cost accrued during the first  $N$  SMDP stages, and  $\sum_{n=0}^{N-1} \mathbb{I}(\mathbf{s}_n \in \mathcal{S}_{\text{comm}})$  is the number of communication states encountered in the SMDP; since a new decision interval is initiated after a communication state, this in turn equals the number of decision intervals. Therefore, after taking the expectation and the limit  $N \rightarrow \infty$ ,  $L_\mu^{(\nu)}$  represents the expected Lagrangian cost per decision interval, as expressed in (8). The right hand expression in (11) follows by noticing that the SMDP achieves a steady-state behavior when  $N \rightarrow \infty$ . Next, we prove that the steady-state probability  $\pi_{\text{comm}}$  is independent of the policy  $\mu$  by detailing the SMDP's stationary expressions. Let  $\pi_{\text{wait}} = 1 - \pi_{\text{comm}}$  be the SMDP steady-state probability of the UAV being in the waiting state. Since the probability of remaining in the waiting state (no request is received) in one SMDP step is  $p_{ww} = e^{-\Lambda\Delta_0}$  and that of moving from a communication state to a waiting state is  $p_{cw} = 1$ , we can write the stationary equations of waiting and communication states of the SMDP as

$$\begin{aligned} \pi_{\text{wait}} &= \pi_{\text{wait}} p_{ww} + \pi_{\text{comm}} p_{cw} = e^{-\Lambda\Delta_0} \pi_{\text{wait}} + \pi_{\text{comm}}; \\ \pi_{\text{comm}} &= \pi_{\text{wait}} (1 - p_{ww}) + \pi_{\text{comm}} (1 - p_{cw}) = (1 - e^{-\Lambda\Delta_0}) \pi_{\text{wait}}; \text{ and} \\ \pi_{\text{wait}} + \pi_{\text{comm}} &= 1, \text{ thereby yielding } \pi_{\text{comm}} = \frac{1 - e^{-\Lambda\Delta_0}}{2 - e^{-\Lambda\Delta_0}}. \end{aligned} \quad (12)$$

Specializing  $\ell_\nu(s; \mathbf{a})$  for the waiting states,  $\ell_\nu(r_U, \theta_U; v_r, \theta_c) = \nu \left( P_{\text{mob}} \left( \sqrt{v_r^2 + r_U^2 \theta_c^2} \right) - P_{\text{avg}} \right) \Delta_0$ ; and for communication states under action  $\xi=0$ ,  $\ell_\nu(r_U, \theta_U, r, \theta; 0, [\cdot]) = \frac{L}{R_{GB}(r)}$ , while under action  $\xi=1$  with trajectory  $\mathbf{p}_U$  of duration  $\Delta$ ,  $\ell_\nu(r_U, \theta_U, r, \theta; 1, \mathbf{p}_U) \triangleq (1 - \nu P_{\text{avg}}) \Delta + \nu \int_0^\Delta P_{\text{mob}}(v_U(\eta)) d\eta$ .

The minimization problem of (8) can then be expressed as the average cost-per-stage problem

$$g(\nu) = \frac{1}{\pi_{\text{comm}}} \min_{\mu} \int_{\mathcal{S}} \Pi_{\mu}(s) \ell_{\nu}(s; \mu(s)) ds, \quad (13)$$

solvable through standard dynamic programming approaches, after discretization of the state and the action spaces, followed by the dual maximization process given by  $\max_{\nu \geq 0} g(\nu)$ .

Since GN transmission requests are uniformly distributed in the circular cell with the BS in the center, the UAV radius information is a sufficient statistic in decision making for a waiting state  $(r_U, \theta_U)$ , which can then be expressed as  $s=r_U \in \mathcal{S}_{\text{wait}}$ . Likewise, for a communication state  $(r_U, \theta_U, r, \theta)$ , only the UAV radius, GN request radius, and the angle  $\psi \in [0, 2\pi)$  between them suffice to characterize the state. Thus communication states can be compactly represented as  $s=(r_U, r, \psi) \in \mathcal{S}_{\text{comm}}$ . A consequence of these sufficient statistics for decision making is that the policy affects the SMDP state transitions (hence steady-state behavior) only through the UAV radial velocity  $v_r$  in the waiting states and the UAV trajectory's target end radius position  $\hat{r}_U$  in the communication states. On the other hand, the angular velocity  $\theta_c$  in the waiting states and the specific trajectory used by the UAV to reach the target end radius  $\hat{r}_U$  in the communication states do not influence state dynamics, but only the instantaneous Lagrangian metric  $\ell_{\nu}$ .

With this observation, let  $O(s) \triangleq v_r \in [-V_{\max}, V_{\max}]$  define the radial velocity policy of the waiting states  $s \in \mathcal{S}_{\text{wait}}$ , specifying the radial velocity component of a waiting action  $a \in \mathcal{A}_{\text{wait}}(s)$ ; let  $U(s) \triangleq \hat{r}_U \in [0, a]$  define the next radius position policy of the communication states  $s \in \mathcal{S}_{\text{comm}}$ , specifying the end radius position of a communication action  $a \in \mathcal{A}_{\text{comm}}(s)$ . Accordingly,  $O$  and  $U$  constitute the SMDP's outer decisions and are the only actions to affect the steady-state distribution, denoted as  $\Pi_{O,U}$  under the outer policy  $(O, U)$ ; thus, (13) can be restated as

$$g(\nu) = \frac{1}{\pi_{\text{comm}}} \min_{O,U} \left[ \int_{\mathcal{S}_{\text{wait}}} \Pi_{O,U}(s) \ell_{\nu}^*(s; O(s)) ds + \int_{\mathcal{S}_{\text{comm}}} \Pi_{O,U}(s) \ell_{\nu}^*(s; U(s)) ds \right], \quad (14)$$

where  $\ell_{\nu}^*$  is the Lagrangian metric optimized with respect to the inner action components not specified by  $O$  and  $U$ . In particular, for a waiting state  $s=r_U$  and corresponding radial velocity action  $O(s)=v_r$ , the inner optimization is performed with respect to the angular velocity  $\theta_c$ ,

$$\ell_{\nu}^*(s; v_r) = \min_{\theta_c} \nu \left( P_{\text{mob}} \left( \sqrt{v_r^2 + r_U^2 \theta_c^2} \right) - P_{\text{avg}} \right) \Delta_0 \quad \text{s.t.} \quad \sqrt{v_r^2 + r_U^2 \theta_c^2} \leq V_{\max}. \quad (15)$$

Since  $\nu \geq 0$ ,  $\Delta_0 > 0$ , and  $P_{\text{avg}}$  are constant, the optimizer  $\theta_c^*$  is the angular velocity minimizing the UAV power consumption for a given UAV radial velocity  $v_r$  and radius  $r_U$ . Due to the

quasi-convex structure of  $P_{\text{mob}}(v)$  [10], it follows that the optimal radial velocity is  $\theta_r = 0$  if  $|v_r| \geq v_{P_{\min}} \triangleq \arg \min P_{\text{mob}}(v)$  (in fact, any radial velocity would unnecessarily increase power consumption), and such that  $\sqrt{v_r^2 + r_U^2 \theta_c^2} = v_{P_{\min}}$  otherwise (so as to match the power minimizing speed). For communication states  $\mathbf{s} = (r_U, r, \psi)$ ,  $\ell_\nu^*(\mathbf{s}; \hat{r}_U)$  is determined by optimizing over the scheduling decision  $\xi \in \{0, 1\}$  and, if  $\xi=1$ , the trajectory  $\mathbf{p}_U$  followed by the UAV, terminating at radius  $\hat{r}_U$ . Formally, let  $\ell_\nu^*(\mathbf{s}; \hat{r}_U, \xi)$  be the optimized metric: for  $\xi=1$  (D&F),

$$\ell_\nu^*(\mathbf{s}; \hat{r}_U, 1) = \min_{\Delta, \mathbf{p}_U, t_p} (1 - \nu P_{\text{avg}}) \Delta + \nu \int_0^\Delta P_{\text{mob}} \left( \sqrt{r'_U(\eta)^2 + r_U^2(\eta) \theta'_U(\eta)^2} \right) d\eta \text{ s.t. C.1, C.2, } (16)$$

$$\mathbf{p}_U(0) = (r_U, 0), \|\mathbf{p}_U(\Delta)\|_2 = \hat{r}_U, \quad (\hat{\text{C.3}})$$

where C.1 reflects the data payload constraints of the D&F protocol, C.2 is the maximum UAV velocity constraint, and  $\hat{\text{C.3}}$  (rewritten version of C.3) enforces the trajectory constraints; for  $\xi=0$  (direct-BS, hence  $\hat{r}_U=r_U$ ),  $\ell_\nu^*(\mathbf{s}; r_U, 0) = \frac{L}{R_{GB}(r)}$ . Hence,  $\ell_\nu^*(\mathbf{s}; r_U)$  is obtained by further minimizing over the scheduling decision  $\xi \in \{0, 1\}$ , yielding the following metric:

$$\ell_\nu^*(\mathbf{s}; \hat{r}_U) = \min_{\xi \in \{0, 1\}} \ell_\nu^*(\mathbf{s}; r_U, \xi) \mathbb{I}(\hat{r}_U = r_U) + \ell_\nu^*(\mathbf{s}; \hat{r}_U, 1) \mathbb{I}(\hat{r}_U \neq r_U). \quad (17)$$

Thus, if the outer decision selects  $U(\mathbf{s})=r_U$ , the inner scheduling decision  $\xi \in \{0, 1\}$  is obtained by greedily minimizing a cost metric that trades off communication delay and energy consumption, i.e., direct transmission to the BS occurs if  $\ell_\nu^*(\mathbf{s}; r_U, 0) < \ell_\nu^*(\mathbf{s}; r_U, 1)$ . Otherwise, the UAV handles the GN request using the D&F protocol, and the inner decision on UAV trajectory greedily minimizes the instantaneous delay-power trade-off, terminating at the target radius  $U(\mathbf{s})=\hat{r}_U$ . We now describe Algorithm 1, which designs the outer policy and computes the average cost-per-stage metric  $g(\nu)$ , along with the average energy- and time-per-stage metrics for a given  $\nu$ , by solving problem (14) via value iteration. Then, we discuss Algorithm 2, which solves the dual maximization problem  $\max_{\nu \geq 0} g(\nu)$  via projected sub-gradient ascent [34]. The source code for these algorithms<sup>2</sup> has been made available in our GitHub repository (MAESTRO-X [2]).

Specifically, in Algorithm 1, lines 2 and 3 compute the inner Lagrangian cost metric optimized with respect to the inner actions—along with the energy and time cost metrics—for all states and outer actions; line 6 computes the value iteration update for waiting states, which accounts for the transition to the waiting state  $r_U + v_r \Delta_0$  (if no transmission request is received, w.p.  $e^{-\Lambda' \Delta_0}$ )

<sup>2</sup>While Algorithm 1 describes continuous state and action spaces, practical implementations constitute their discretized versions.

---

**Algorithm 1** Value Iteration:  $(O^*, U^*, g(\nu), \bar{E}, \bar{T}) = \text{VITER}(\nu)$ 


---

- 1: **Initialization:**  $i=0$ ; the value function  $V_i(\mathbf{s})=0$ , total energy cost  $E_i(\mathbf{s})=0$ , and total time cost  $T_i(\mathbf{s})=0$ ,  $\forall \mathbf{s} \in \mathcal{S}$ ; stop criterion  $\delta$ .
- 2: **Inner optimization in waiting states:**  $\forall \mathbf{s} \in \mathcal{S}_{\text{wait}}, \forall v_r \in [-V_{\max}, V_{\max}]$ , calculate  $\ell_\nu^*(\mathbf{s}; v_r)$  as in (15), with minimizer  $\theta_c^*$ ; compute energy cost  $e^*(\mathbf{s}; v_r) = E(\mathbf{s}; v_r, \theta_c^*)$  and time cost  $t^*(\mathbf{s}; v_r) = T(\mathbf{s}; v_r, \theta_c^*)$ .
- 3: **Inner optimization in communication states:**  $\forall \mathbf{s} \in \mathcal{S}_{\text{comm}}, \forall \hat{r}_U \in [0, a]$ , calculate  $\ell_\nu^*(\mathbf{s}; \hat{r}_U)$  as in (16) and (17) using Algorithm 3, with minimizer  $(\xi^*, \hat{\theta}_U^*, \mathbf{p}_U^*, \Delta^*)$ ; compute energy cost  $e^*(\mathbf{s}; \hat{r}_U) = \xi^* E(\mathbf{s}; \hat{r}_U, \hat{\theta}_U^*, \mathbf{p}_U^*)$  and time cost  $t^*(\mathbf{s}; \hat{r}_U) = \xi^* T(\mathbf{s}; \hat{r}_U, \hat{\theta}_U^*, \mathbf{q}_U^*)$ .
- 4: **repeat**
- 5:   **for** each  $\mathbf{s}=r_U \in \mathcal{S}_{\text{wait}}$  **do**
- 6:      $V_{i+1}(\mathbf{s}) \leftarrow \min_{v_r \in [-V_{\max}, V_{\max}]} [\ell_\nu^*(\mathbf{s}; v_r) + e^{-\Lambda' \Delta_0} V_i(r_U + v_r \Delta_0) + (1 - e^{-\Lambda' \Delta_0}) \int_0^{2\pi} \frac{1}{2\pi} \int_0^a \frac{2r'}{a^2} V_i(r_U + v_r \cdot \Delta_0, r', \psi') dr' d\psi']$ ,  
      where  $O_{i+1}(\mathbf{s})$  is the arg min.
- 7:      $E_{i+1}(\mathbf{s}) \leftarrow e^*(\mathbf{s}; O_{i+1}(\mathbf{s})) + e^{-\Lambda' \Delta_0} E_i(r_U + O_{i+1}(\mathbf{s}) \Delta_0) + (1 - e^{-\Lambda' \Delta_0}) \int_0^{2\pi} \frac{1}{2\pi} \int_0^a \frac{2r'}{a^2} E_i(r_U + O_{i+1}(\mathbf{s}) \Delta_0, r', \psi') dr' d\psi'$ .
- 8:      $T_{i+1}(\mathbf{s}) \leftarrow t^*(\mathbf{s}; O_{i+1}(\mathbf{s})) + e^{-\Lambda' \Delta_0} T_i(r_U + O_{i+1}(\mathbf{s}) \Delta_0) + (1 - e^{-\Lambda' \Delta_0}) \int_0^{2\pi} \frac{1}{2\pi} \int_0^a \frac{2r'}{a^2} T_i(r_U + O_{i+1}(\mathbf{s}) \Delta_0, r', \psi') dr' d\psi'$ .
- 9:   **end for**
- 10:   **for** each  $\mathbf{s}=(r_U, r, \psi) \in \mathcal{S}_{\text{comm}}$  **do**
- 11:      $V_{i+1}(\mathbf{s}) \leftarrow \min_{\hat{r}_U \in [0, a]} [\ell_\nu^*(\mathbf{s}; \hat{r}_U) + V_i(\hat{r}_U)]$ , where  $U_{i+1}(\mathbf{s})$  is the arg min.
- 12:      $E_{i+1}(\mathbf{s}) \leftarrow e^*(\mathbf{s}; U_{i+1}(\mathbf{s})) + E_i(U_{i+1}(\mathbf{s})); T_{i+1}(\mathbf{s}) \leftarrow t^*(\mathbf{s}; U_{i+1}(\mathbf{s})) + T_i(U_{i+1}(\mathbf{s}))$ .
- 13:   **end for**
- 14:    $\forall \mathbf{s}$ , calculate the stopping criterion metric  $H(\mathbf{s}) = V_{i+1}(\mathbf{s}) - V_i(\mathbf{s})$ ;  $i \leftarrow i+1$ .
- 15: **until**  $\max_{\mathbf{s} \in \mathcal{S}} H(\mathbf{s}) - \min_{\mathbf{s} \in \mathcal{S}} H(\mathbf{s}) < \delta$ .
- 16: Approximate  $[g(\nu); \bar{E}; \bar{T}] \approx \frac{1}{\pi_{\text{comm}}} \frac{1}{i} [V_i(\mathbf{s}); E_i(\mathbf{s}); T_i(\mathbf{s})]$ , for some arbitrary  $\mathbf{s} \in \mathcal{S}$ .
- 17: **return**  $O^*(\mathbf{s}) = O_i(\mathbf{s}), \forall \mathbf{s} \in \mathcal{S}_{\text{wait}}, U^*(\mathbf{s}) = U_i(\mathbf{s}), \forall \mathbf{s} \in \mathcal{S}_{\text{comm}}, g(\nu), \bar{E}$ , and  $\bar{T}$ .

---

**Algorithm 2** Projected Sub-gradient Ascent: PSGA()

---

- 1: **Initialization:**  $k=0$ ; dual variable  $\nu_0 \geq 0$ ; step-size  $\{\rho_j = \frac{\rho_0}{(j+1)}, j \geq 0\}$ ;  $g_{-1} = \infty$ .
- 2: **for**  $k=0, 1, \dots$  **do**
- 3:   Determine  $(O_k^*, U_k^*, g_k, \bar{E}_k, \bar{T}_k) = \text{VITER}(\nu_k)$  via Algorithm 1.
- 4:   **if**  $|g_k - g_{k-1}| < \epsilon_{DI}$ ;  $\bar{E}_k - P_{\text{avg}} \bar{T}_k < \epsilon_{PF}$ ;  $\nu_k |\bar{E}_k - P_{\text{avg}} \bar{T}_k| < \epsilon_{CS}$  **then**
- 5:     **return:** optimal outer policy  $(O_k^*, U_k^*)$ ;
- 6:   **else**
- 7:     Update  $\nu_{k+1} = \max \{\nu_k + \rho_k (\bar{E}_k - P_{\text{avg}} \bar{T}_k), 0\}$ ;  $k \leftarrow k+1$ .
- 8:   **end if**
- 9: **end for**

---

and to a communication state  $(r_U + v_r \Delta_0, r', \psi')$  (if a transmission request is received in position  $(r', \psi')$ , with probability density function  $\frac{r'}{\pi a^2}$ ); line 11 computes the value iteration update for communication states which—for end radius  $\hat{r}_U$ —accounts for the transition to the waiting state  $\hat{r}_U$ , w.p. 1; lines 7, 8, 12, and 13 update the corresponding total energy and time costs; line 17 estimates the values of the average cost-, energy-, and time-per-stage metrics.

In Algorithm 2, line 1 initializes the dual variable and a sequence of step-sizes used for projected dual sub-gradient ascent; line 3 calls the value iteration of Algorithm 1 using the current dual variable value  $\nu_k$ , and outputs the optimal radial velocity (for waiting states) and next radius position (for communication states) policies, as well as the average cost-, energy-, and time-per-stage metrics; line 4 monitors convergence, by checking the change in the average

cost-per-stage value, as well as primal feasibility and complementary slackness, with respect to the average UAV power constraint; line 7 updates the value of the dual variable in the direction of its sub-gradient and projects the value to the non-negative range to ensure dual feasibility. We are left with the trajectory design (line 3 of Algorithm 1), which is carried out in the next section via the Hierarchical Competitive Swarm Optimization (HCSO) algorithm.

#### IV. MAESTRO: TRAJECTORY DESIGN VIA HIERARCHICAL CSO

In this section, we design the UAV trajectory during the D&F protocol by solving (16). The literature solves problems like (16) with SCA (see [10], [16], [35]), in which the non-convex objective and constraints are approximated by convex functions around a local point; the approximated problem is solved with convex optimization methods, the minimizers become the next local point, convex approximations are made about the new point, and the process repeats until convergence. However, SCA has major drawbacks: 1) it is computationally slow, as shown numerically in Sec. VI; 2) its QoS constraints on the GN data payload require additional approximations, which are not employed in our model, and these preclude the adoption of rate adaptation at the transmitters to efficiently exploit the A2G channel characteristics.

Within MAESTRO, to overcome these challenges, we propose a framework based on Competitive Swarm Optimization (CSO) [24]. Since the design space involves optimizing over infinitely many variables  $(r_U(t), \theta_U(t))$ , we simplify the continuous UAV trajectory into a finite sequence of way-points connected by straight lines at constant velocity. However, a direct application of CSO suffers from poor convergence because solution spaces of such problems tend to increase exponentially with dimensionality [36]. We address this weakness by proposing a Hierarchical CSO (HCSO) algorithm, in which a sequence of CSO problems is solved: initially, CSO produces a low-resolution trajectory; the optimized trajectory is then used to create a higher-resolution trajectory via interpolation, which is further optimized with CSO; and so on. This sequence of CSO problems generate trajectories of increasing resolution and helps alleviate the poor performance achieved when optimizing the high-resolution trajectory directly. First, we describe the meta-heuristic trajectory; then, we design a hierarchical wrapper enveloping CSO and show how high-dimensional trajectories leverage low-dimensional ones to boost accuracy.

**Formulation of the Meta-Heuristic UAV Trajectory:** To define the UAV meta-heuristic trajectory, we use Cartesian coordinates instead of polar coordinates, for easier evaluation of the data payload constraints. For a state  $\mathbf{s} = (r_U, r, \psi) \in \mathcal{S}_{\text{comm}}$  and a next radius position  $U(\mathbf{s}) = \hat{r}_U$ , we encode a trajectory as a sequence of way-points and UAV velocities. In particular, we define a meta-

heuristic trajectory of  $\mathbf{p}_U$  by a sequence of  $M$  trajectory segments,  $\{\mathbf{x}_m\}_{m=0}^M = \{(x_m, y_m)\}_{m=0}^M$ . The UAV trajectory begins at  $\mathbf{x}_0$  and ends at  $\mathbf{x}_M$ ; the UAV traverses each straight trajectory segment  $\Psi_m \triangleq \mathbf{x}_{m+1} - \mathbf{x}_m$  with velocity  $v_m \in [V_{\text{low}}, V_{\text{max}}]$ , with minimum velocity  $V_{\text{low}} \ll V_{\text{max}}$  introduced to ensure well-defined durations; the sequences of trajectory way-points  $\mathbf{p} \triangleq [\mathbf{x}_1, \dots, \mathbf{x}_M]^T$  and velocities  $\mathbf{v} \triangleq [v_0, \dots, v_{M-1}]^T$  are the optimization variables. For  $M$  trajectory segments, the first and second  $\frac{M}{2}$  segments correspond to the two-phase D&F protocol. As the throughput equations are angular-invariant, without loss of generality, the initial UAV location is set to  $\mathbf{x}_0 = (r_U, 0)$  and the GN request location to  $\mathbf{x}_G \triangleq (r \cos \psi, r \sin \psi)$ . Since the number of bits communicated (C.1) during each trajectory segment cannot be computed in closed-form, we approximate them for each trajectory segment numerically. Specifically, between subsequent way-points,  $\mathbf{x}_m$  and  $\mathbf{x}_{m+1}$  with velocity  $v_m$ , we generate a sequence of  $n_{\text{res}}$  evenly-spaced points with a sufficiently high resolution, i.e.,  $\{\mathbf{x}_k^{\text{new}}\}_{k=0}^{n_{\text{res}}}$ ; the time durations between the sequence of points,  $\{t_k^{\text{new}}\}_{k=0}^{n_{\text{res}}-1}$ , and expected throughput at each point,  $\{R_k^{\text{new}}\}_{k=0}^{n_{\text{res}}-1}$ , then determine the number of bits communicated for the  $m$ th segment by calculating  $F_m \triangleq \sum_{k=0}^{n_{\text{res}}-1} t_k^{\text{new}} R_k^{\text{new}}$ . With this numerical approximation, the problem of  $\ell_\nu^*(\mathbf{s}; U(\mathbf{s}))$  to be solved with HCSO is

$$(\mathbf{P.0}) \quad \min_{\mathbf{p}, \mathbf{v}} \quad (1 - \nu P_{\text{avg}}) \sum_{m=0}^{M-1} \frac{\|\Psi_m\|_2}{v_m} + \nu \sum_{m=0}^{M-1} \frac{\|\Psi_m\|_2}{v_m} P_{\text{mob}}(v_m) \quad (18)$$

$$\text{s.t. } h_i(\mathbf{p}, \mathbf{v}) \triangleq L - \sum_{m=\frac{M}{2}(i-1)}^{\frac{M}{2}(i-1)+\frac{M}{2}-1} F_m \leq 0, \quad i = 1 \text{ and } 2, \quad (\tilde{C}.1)$$

$$\mathbf{x}_0 = (r_U, 0), \quad \mathbf{x}_M = \hat{r}_U \frac{\mathbf{x}_{M-1}}{\|\mathbf{x}_{M-1}\|_2} \mathbb{I}(\|\mathbf{x}_{M-1}\|_2 \neq 0) + (\hat{r}_U, 0) \mathbb{I}(\|\mathbf{x}_{M-1}\|_2 = 0), \quad (\tilde{C}.3)$$

where  $\frac{\|\Psi_m\|_2}{v_m}$  is the duration of the  $m$ th trajectory segment,  $\tilde{C}.1$  (rewritten version of C.1) enforces the data payload constraint, and  $\tilde{C}.3$  (rewritten version of C.3) enforces the trajectory constraints. To satisfy the end radius constraint, the final trajectory way-point  $\mathbf{x}_M$  is determined by projecting the penultimate way-point  $\mathbf{x}_{M-1}$  to the circle at radius  $\hat{r}_U$ , as expressed in  $\tilde{C}.3$ . We solve  $(\mathbf{P.0})$  with HCSO, as it promises to provide results in a computationally tractable manner.

**Solution of  $(\mathbf{P.0})$  via the HCSO Method:** The HCSO method is executed by converting the constrained problem  $(\mathbf{P.0})$  into an unconstrained one and solving a sequence of unconstrained problems with CSO. We enforce the data payload constraints by penalizing constraint violations with a particular solution: if the UAV does not decode (or forward) its data payload at the end of either phase, then it flies along the circumference of a circle (radius  $r_{\min} > 0$ , small) around the

current position with its power-minimizing velocity ( $v_{P_{\min}}=22 \text{ m s}^{-1}$  [10]) until it fully receives (or transmits) this data payload. Thus, the penalized objective function is given as

$$\begin{aligned}\hat{f}(\mathbf{p}, \mathbf{v}) &\triangleq (1-\nu P_{\text{avg}}) \sum_{m=0}^{M-1} \frac{\|\Psi_m\|_2}{v_m} + \nu \sum_{m=0}^{M-1} \frac{\|\Psi_m\|_2}{v_m} P_{\text{mob}}(v_m) + (1-\nu P_{\text{avg}})(\hat{t}_{P,1} + \hat{t}_{P,2}) + \nu(\hat{E}_{P,1} + \hat{E}_{P,2}); \\ \hat{t}_{P,1} &\triangleq \frac{h_1(\mathbf{p}, \mathbf{v}) \mathbb{I}(h_1(\mathbf{p}, \mathbf{v}) > 0)}{\bar{R}_{GU} (\|\mathbf{x}_{M/2} - \mathbf{x}_G\|_2)}; \quad \hat{t}_{P,2} \triangleq \frac{h_2(\mathbf{p}, \mathbf{v}) \mathbb{I}(h_2(\mathbf{p}, \mathbf{v}) > 0)}{\bar{R}_{UB} (\|\mathbf{x}_M\|_2)}; \quad \hat{E}_{P,i} \triangleq P_{\text{mob}}(v_{P_{\min}}) \hat{t}_{P,i},\end{aligned}$$

where  $\hat{t}_{P,i}$  and  $\hat{E}_{P,i}$  correspond to the time and energy penalties for the additional flying involved in finishing the data communication during the decode and forward phases ( $i=1$  and  $2$ ).

We begin by randomly initializing a population of  $M$ -segment trajectory solutions (particles). Specifically, in the first call to the CSO algorithm [24] inside the HCSO framework, we initialize  $N$  trajectory particles  $\mathbf{p}_{1:N} \triangleq \mathbf{p}_1, \dots, \mathbf{p}_N$ , with particle velocities  $\mathbf{u}_{1:N} \triangleq \mathbf{u}_1, \dots, \mathbf{u}_N$ , and  $N$  UAV velocity particles  $\mathbf{v}_{1:N} \triangleq \mathbf{v}_1, \dots, \mathbf{v}_N$ , with particle velocities  $\mathbf{w}_{1:N} \triangleq \mathbf{w}_1, \dots, \mathbf{w}_N$ , imposing no trajectory structure in the first iteration. After the first CSO iteration, the resulting trajectory is interpolated to form a reference  $2M$ -segment trajectory,  $\tilde{\mathbf{p}} = [\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{M-1}]^T$  and  $\tilde{\mathbf{v}} = [\tilde{v}_0, \dots, \tilde{v}_{M-1}]^T$ , with  $M \leftarrow 2M$ , so as to increase the resolution for the next iteration. The new population size is then reduced,  $N \leftarrow N - N_{\text{red}}$ , to lower the computational burden of CSO, and a new set of trajectories is generated randomly in a neighborhood around the reference trajectory. More precisely, the  $m$ th way-point,  $\mathbf{x}_m = (x_m, y_m)$ , of a new trajectory is generated as  $\mathbf{x}_m = \tilde{\mathbf{x}}_m + (\chi_m, \zeta_m)$ , where the localized randomness in the new way-points is modeled using zero-mean Gaussian random variables,  $\chi_m, \zeta_m \sim \mathcal{N}(0, \varsigma (\|\tilde{\mathbf{x}}_{m+1} - \tilde{\mathbf{x}}_m\|^2 + \|\tilde{\mathbf{x}}_{m-1} - \tilde{\mathbf{x}}_m\|^2))$ , with  $\varsigma > 0$  as a scaling factor, and the variance determined by the spread between neighboring reference trajectory way-points, incorporated due to the empirical observation that areas with clustered UAV way-points are the regions where the objective function  $\hat{f}(\mathbf{p}, \mathbf{v})$  is sensitive to large variations. Likewise, the UAV velocity of the  $m$ th new trajectory segment is generated by  $v_m = [\tilde{v}_m + \varkappa_m]^{[V_{\text{low}}, V_{\text{max}}]}$  where  $\varkappa_m \sim \mathcal{N}(0, \varepsilon (V_{\text{max}} - V_{\text{low}})^2)$  induces randomness in the new velocity,  $\varepsilon > 0$  is a scaling factor, and  $[\cdot]^{[V_{\text{low}}, V_{\text{max}}]} = \max \{ \min \{ \cdot, V_{\text{max}} \}, V_{\text{low}} \}$  projects velocity  $v_m$  to the feasible set. The variance is chosen due to the observation that the UAV velocities exhibit consistent convergence with CSO much faster than the trajectory way-points and are less sensitive to random initializations. This structure of HCSO is depicted in Algorithm 3.

Line 1 initializes  $N$  random  $M$ -segment UAV trajectories; line 3 obtains an  $M$ -segment trajectory with CSO [24]; line 4 doubles the trajectory resolution and interpolates the CSO

---

**Algorithm 3** HCSO Algorithm for  $\ell_\nu^*(s; U(s))$ 


---

```

1: Randomly initialize UAV way-points  $\mathbf{p}_{1:N}$  and speeds  $\mathbf{v}_{1:N}$ .
2: while  $M \leq M_{\max}$  do
3:   Obtain  $M$ -segment trajectory:  $(\mathbf{p}^*, \mathbf{v}^*) = \text{CSO}(\mathbf{p}_{1:N}, \mathbf{v}_{1:N}, N, M)$  (see [24]).
4:   Increase  $M \leftarrow 2M$ ; interpolate to form reference trajectory:  $(\tilde{\mathbf{p}}, \tilde{\mathbf{v}}) = \text{interp}(\mathbf{p}^*, \mathbf{v}^*, M)$ .
5:   Reduce swarm size  $N \leftarrow N - N_{\text{red}}$ .
6:   for  $n_1 = 1, 2, \dots, N$  do
7:     for  $n_2 = 1, 2, \dots, M - 1$  do
8:       Extract  $\tilde{\mathbf{x}}_{n_2} = (\tilde{x}_{n_2}, \tilde{y}_{n_2})$  from reference  $\tilde{\mathbf{p}}$  and generate  $\mathbf{x}_{n_2}$  of  $\mathbf{p}_{n_1}$  via  $\mathbf{x}_{n_2} = \tilde{\mathbf{x}}_{n_2} + (\chi_{n_2}, \zeta_{n_2})$ .
9:     end for
10:    for  $n_3 = 0, \dots, M - 1$  do
11:      Extract  $\tilde{v}_{n_3}$  from reference  $\tilde{\mathbf{v}}$  and generate  $v_{n_3}$  of  $\mathbf{v}_{n_1}$  via  $v_{n_3} = [\tilde{v}_{n_3} + \varkappa_m]^{[V_{\text{low}}, V_{\text{max}}]}$ .
12:    end for
13:    Update new swarm particle  $\mathbf{p}_{n_1} \leftarrow [\mathbf{x}_1, \dots, \mathbf{x}_{M-1}]^T$  and  $\mathbf{v}_{n_1} \leftarrow [v_0, \dots, v_{M-1}]^T$ .
14:   end for
15: end while

```

---

output to a  $2M$ -segment reference trajectory; and lines 8 and 11 generate random trajectories in a neighborhood of the reference trajectory,  $(\tilde{\mathbf{p}}, \tilde{\mathbf{v}})$ . This process repeats until the trajectory resolution equals a desired value. The CSO algorithm, called in line 3, follows [24] and involves the following operations. During the  $k$ th iteration inside CSO, the  $N$  particles are randomly grouped into  $\frac{N}{2}$  pairwise competitions. For both members of a pair,  $\hat{f}(\mathbf{p}, \mathbf{v})$  is calculated; the winner of the competition is passed onto the  $(k+1)$ th iteration of CSO, and the loser is modified by learning from the winner, as detailed by the loser particle and velocity update equations in [24]. Next, we discuss the extension of our single-relay specialization to UAV swarms.

## V. MAESTRO-X: AN EXTENSION TO UAV SWARMS VIA SUPPLEMENTARY HEURISTICS

In this section, we extend the optimal scheduling and trajectory optimization framework developed in Secs. III and IV (MAESTRO) to our generalized wireless network model of  $N_U$  UAV relays in the swarm. MAESTRO's eXtension, termed MAESTRO-X, constitutes augmentations to the multiscale optimal waiting and communication policy obtained via SMDP value iteration. Here, we also allow the UAVs and the BS to queue requests, and the scheduling decisions in this network are driven by a modified Lagrangian metric that takes into account the queuing latencies experienced at the BS and the UAVs. Depicting an example scenario of serving data traffic generated by an aggregation of soil sensors in precision agriculture, Fig. 3 illustrates the control flow of MAESTRO-X—in particular, the operations of  $\text{UAV}_2$ . Along with a fully-connected distributed mesh network (employing the designated channels at the band-edges of the allocated spectrum) overlaid over the BS and the UAVs for command-and-control, enhancements to MAESTRO include spread maximization and conflict resolution: these are discussed next.

**Spread Maximization:** We motivate the need for spread maximization by noting that the outer action of MAESTRO's optimal waiting policy constitutes the UAV radial velocity component, which is symmetric with respect to the clockwise and counter-clockwise angular movements of

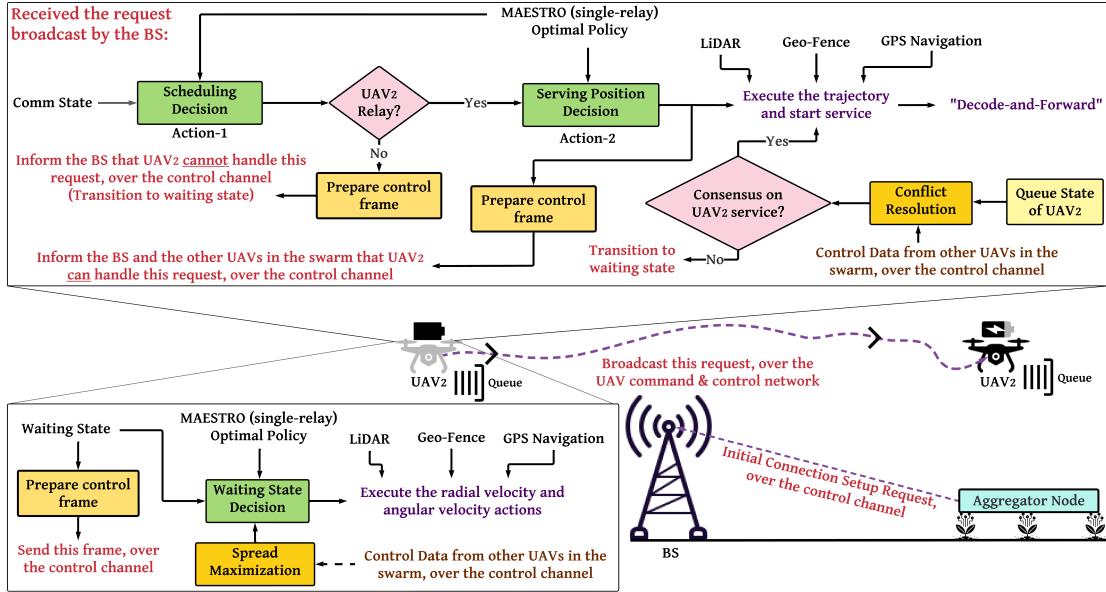


Fig. 3: The study of a wireless network augmented by a swarm of UAV relays in precision agriculture: an illustration outlining the sequence of operations under MAESTRO-X that occur at UAV<sub>2</sub> in the swarm.

the UAV. In multi-UAV settings, we can leverage this symmetry to efficiently prime and position the idle UAVs for potential new relay requests originating in the cell, which is the objective of our spread maximization algorithm. Under this context, a UAV in the swarm, waiting in a particular radius level with the optimal radial velocity action, and with the magnitude of its angular motion obtained from the optimal inner decision at this radius level (15), determines the direction of this angular motion (clockwise or counter-clockwise) based on our spread maximization heuristic—wherein each UAV relay in the waiting state, executes either positive (counter-clockwise) or negative (clockwise) angular movements in order to maximize the minimum distance among them. These coordinated movements among the UAVs in the swarm are made possible through periodic exchanges of control messages, whose frame structure is illustrated in Fig. 4.

With periodic exchanges of control frames (synchronized reporting period  $\delta > 0$ , small), UAV  $i$  waiting in the radius level  $r$  with optimal radial velocity  $v_r^*$  and optimal angular velocity magnitude  $\theta_c^*$ , parses the state flag and GPS event fields in the received control frames, and constructs a local peer list  $\mathcal{L}$  of other UAVs in the waiting state. Let the current angular position of UAV  $i$  be  $\theta_i \in [0, 2\pi]$ . First, UAV  $i$  determines the index of its closest peer (in the angular dimension), i.e.,  $j^* = \arg \min_{j \in \mathcal{L}} |\theta_i - \theta_j|$ . Let the current angular coordinate of this UAV  $j^*$  be  $\theta_{j^*} \in [0, 2\pi]$ . Next, UAV  $i$  determines its direction of angular motion as  $\rho^* = \arg \max_{\rho \in \{+1, -1\}} |[\theta_i + \delta \rho \theta_c^*]^{[0, 2\pi]} - \theta_{j^*}|$ , wherein UAV  $i$  is moving away from its closest UAV in an attempt to spread out such that it is suitably positioned for potential new requests from this region of the cell. Then, UAV  $i$  executes

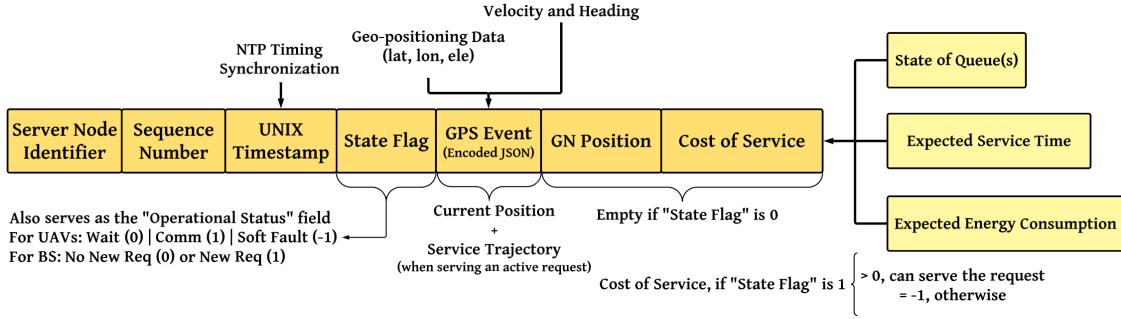
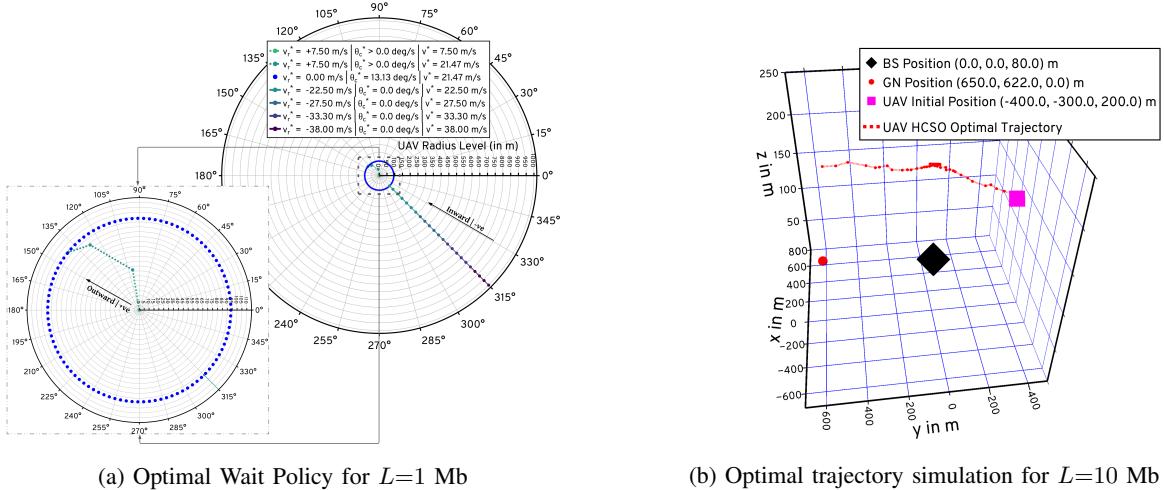


Fig. 4: The structure of a control frame for inter-UAV and BS-UAV coordination in MAESTRO-X.

this angular motion until new control frames (containing updated positions) are received from its peers (at the end of the synchronized reporting period) or upon reception of a new uplink transmission request from a GN in the cell, after which it transitions to the communication state and the consensus-driven conflict-resolution process (which is discussed next) begins.

**Consensus-driven Conflict Resolution:** In our single-relay formulation discussed in Sec. 3, the scheduling action involved a comparison between the Lagrangian cost of direct transmission to the BS and that of relayed service through the UAV. However, while extending to UAV swarms via policy replication and with the incorporation of queuing dynamics, this single-agent scheduling decision has to be enhanced to 1) resolve conflicts among server nodes (BS and UAVs), when two or more of them decide to serve a new GN request (according to their local execution of MAESTRO) and hence facilitate a consensus on the best possible node to serve the GN; and 2) account for the queue wait times experienced by the GN at each potential server node. Accordingly, the single-agent comparison operation for scheduling is scaled to multiple UAVs and modified to incorporate the impact of queuing dynamics: this augmentation is termed consensus-driven conflict resolution and is driven by a cost-of-service metric defined below.

For a request at  $(r, \theta)$ ,  $\frac{L}{R_{GB}(r)} + t_{Q|BS}$  at the BS and  $\tilde{f}(\mathbf{p}_i^*, \mathbf{v}_i^*) = \hat{f}(\mathbf{p}_i^*, \mathbf{v}_i^*) + (1 - \nu P_{avg})t_{Q|i}$  at an available UAV  $i$ , constitutes the comparison cost metric, where  $t_{Q|BS}$  denotes the waiting time in a BS queue and  $t_{Q|i}$  denotes the wait time for the request in the UAV  $i$  queue. Specifically, this queue wait time includes the time needed to serve the requests already in the queue. Moreover, note that in our construction—corresponding to the Poisson arrival process and the number of orthogonal channels at the BS and the UAVs—the BS models an M/G/ $N_B$  queuing system while the UAVs model an M/G/1 queue each. When a new request originates in the cell, the UAVs already serving a GN continue to do so, i.e., they do not participate in the consensus-driven conflict resolution process. On the other hand, UAVs in the waiting states transition into their respective communication states: let these UAVs constitute the set  $\mathcal{L}'$ . The BS along with these



(a) Optimal Wait Policy for  $L=1$  Mb

(b) Optimal trajectory simulation for  $L=10$  Mb

Fig. 5: (a) The execution of the optimal waiting state policy at a UAV for  $L=1$  Mb with  $P_{\text{avg}}=1.2$  kW; (b) The optimal trajectory simulation of UAV<sub>2</sub> under the MAESTRO-X framework for  $L=10$  Mb and  $P_{\text{avg}}=1.2$  kW.

relays are deemed to be available. These available server nodes exchange collaboration messages over the command-and-control network (see Fig. 4). These messages comprise the cost-of-service metric  $\tilde{f}(\mathbf{p}_i^*, \mathbf{v}_i^*)$  which enables the available nodes to arrive at a consensus on the best choice to serve the request, i.e., if  $\frac{L}{R_{GB}(r)} + t_{Q|BS} < \tilde{f}(\mathbf{p}_i^*, \mathbf{v}_i^*), \forall i \in \mathcal{L}'$ , then direct transmission to the BS is scheduled; else, the GN request is relayed through UAV  $i^* = \arg \min_{i \in \mathcal{L}'} \tilde{f}(\mathbf{p}_i^*, \mathbf{v}_i^*)$ .

## VI. SIMULATION SETUP AND NUMERICAL EVALUATIONS

In our numerical evaluations, we use a channel bandwidth of  $B = 5$  MHz; for all links, NLoS attenuation constant  $\kappa = 0.2$ , 1-meter reference SNR  $\frac{\beta_0 P}{\sigma^2 \Gamma} = 40$  dB, LoS path-loss exponent  $\alpha = 2$ , NLoS path-loss exponent  $\tilde{\alpha} = 2.8$ , Rician  $K$ -factor parameters  $k_1 = 1$  and  $k_2 = 0.05$  [14], and LoS probability parameters  $z_1 = 9.61$  and  $z_2 = 0.16$  [37]; UAV height  $H_U = 200$  m; BS antenna height  $H_B = 80$  m; maximum UAV velocity  $V_{\text{max}} = 55$  m s<sup>-1</sup>; and cell radius  $a = 1000$  m. The UAV power consumption model uses the relationship and parameters detailed in [10]. To solve (14), we discretize the state and action spaces and apply linearly-interpolated value iteration. We discretize the states with  $N_{\text{sp}} = 25$  equispaced radii values, with 12 GNs in each level; and  $R_{\text{sp}} = 25$  equispaced radial velocity actions for  $v_r \in \{-V_{\text{max}}, \dots, V_{\text{max}}\}$ . Lastly,  $\Delta_0$  is chosen to satisfy  $e^{-\Lambda' \Delta_0} = 0.93$ , such that it is unlikely to receive two or more GN requests in  $\Delta_0$ .

First, to analyze the optimal waiting behavior of a UAV relay in our control framework, we fix  $P_{\text{avg}}=1.2$  kW and  $L=1$  Mb. Fig. 5a shows that the UAV moves to and waits ( $v_r^*=0$ ) by flying around an optimal radius level ( $r_0^* \approx 95$  m) to address two considerations: to be well-positioned for future requests and to fly at the power-minimizing velocity so as to reduce its energy consumption. Moreover, angular velocity optimization adheres to the observations in [10]

about optimizing towards  $P_{\min}$  (corresponding to  $V = 22 \text{ m s}^{-1}$ ). Next, Fig. 5b depicts a simulated execution of the optimal trajectory of  $\text{UAV}_2$  (obtained via HCSO in Sec. IV) under  $L=10 \text{ Mb}$  traffic payloads and an average power constraint of  $P_{\text{avg}}=1.2 \text{ kW}$ , when an active GN request at  $(650, 622, 0)$  in the cell is scheduled to be served by the UAV—via the D&F protocol detailed in Sec. 3—with this trajectory beginning at the initial UAV position of  $(-400, -300, 200)$ .

Second, we evaluate the delay-power trade-off of MAESTRO-X and compare with state-of-the-art algorithms: Successive Convex Approximation (SCA) [10], Constrained SCA with Alternating Direction Method-of-Multipliers (CSCA-ADMM) [16], and Double Deep Q-Networks with Combined Experiential Replay (DDQN-CER) [19]. As depicted in Figs. 6a and 6b, for uplink transmission requests of size  $L=1 \text{ Mb}$  from the 300 GNs in the cell, we observe the following improvements in performance over custom network deployment heuristics and state-of-the-art frameworks, averaged over 10,000 requests with a Poisson arrival rate of 1 request every 60 s. Considering only direct transmissions to a 10-channel OFDMA BS at the cell center, we find a significant reduction in the average communication delay experienced by the GNs in the cell, by employing UAVs to relay data traffic. Also, we observe that employing dynamic UAVs with optimized trajectories demonstrate lower service delays compared to static relay deployments—specifically, for a swarm of 3 UAV relays, with  $L=1 \text{ Mb}$  and a per-UAV power consumption of  $1.37 \text{ kW}$ , our solution services GN requests  $12.5\times$  faster than a static deployment of 3 UAVs positioned equidistant from the cell center. With CVXPY implementations of joint multi-agent SCA strategies [10], [16], we note that our control system with 1 UAV relay exceeds the QoS performance offered by 3 relays under these SCA approaches. Also, at  $P_{\text{avg}}=1.1 \text{ kW}$ , MAESTRO-X demonstrates  $3.25\times$  faster service times relative to the DDQN solution in [19].

Additionally, Fig. 6b illustrates the components of QoS latencies experienced by GNs in our network: the transmission delay (in green) which constitutes the execution of either direct-BS transmission or the D&F protocol of relayed-service through a UAV, and the amount of time spent waiting (in red) in one of  $N_B=10$  BS queues or a UAV queue. Expectedly, under a Join-Fastest-Queue heuristic, the BS-only M/G/ $N_B$  queuing system encounters zero queue wait times due to the availability of multiple queues; similar trends are observed across UAV-assisted implementations when the number of UAVs in the swarm increases. Moreover, we observe a direct correlation between the transmission times and the queue wait times, i.e., for 1-UAV networks with high transmission times (e.g., static UAVs, SCA [10], DDQN-CER [19]), requests experience long delays waiting to be picked up by this single UAV queue under relayed-service.

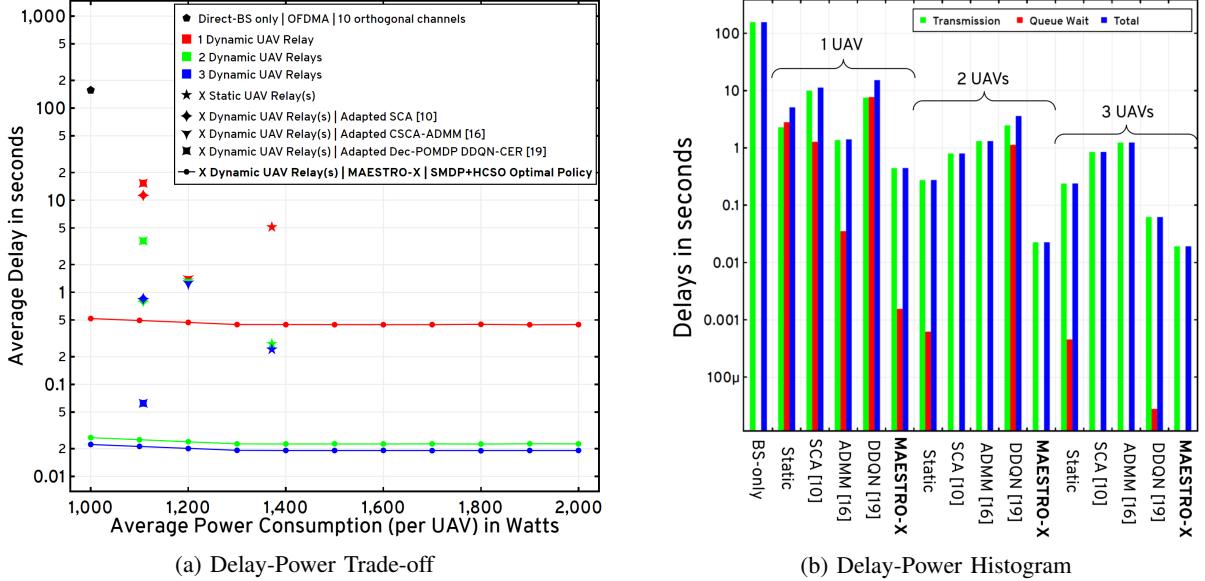


Fig. 6: The traces (a) and histogram (b) constituting the average service latencies versus the average power constraint for MAESTRO-X under  $L=1$  Mb: Comparisons with custom deployment heuristics and with the state-of-the-art.

Third, as shown in Fig. 7a, bench-marking our MAESTRO-X framework against both model-based [16] and model-free [19] solutions in the state-of-the-art, we observe that the decoupled policy replication across the swarm—with suitable supplementary heuristics—ensures that our solution constitutes a practically feasible formulation even with higher-degrees of scaling in the number of UAVs in the swarm. The CSCA-ADMM approach in [16] and the DDQN-CER approach in [19] involve prohibitively large policy convergence times when the number of the UAVs in the swarm is scaled beyond  $N_U=5$  and  $N_U=6$ , respectively. All implementations are in Python, and this bench-marking is performed on a compute node with  $2 \times$  64-core AMD EPYC Milan 7763 CPUs,  $16 \times$  64 GB DDR4 memory, and  $4 \times$  NVIDIA A100 GPUs with 40 GB VRAM each. The CSCA-ADMM solution in [16] suffers poor bench-marking performance relative to MAESTRO-X due to a joint multi-UAV construction involved in its CVXPY-SCS implementation; the DDQN-CER solution in [19] involves larger policy convergence times due to an underlying model-free formulation that employs combined MDP state and action spaces.

Finally, we discuss the emulations of our optimal communication state policy on the NSF AERPAW platform: the setup of this emulation process on the advanced wireless platform includes provisioning both fixed (GN/BS) and portable node (UAV) resources, OFDM PHY at all radios equipped with adaptive modulation and coding driven by SNR feedback, and the integration of our optimal control framework with the underlying radio and vehicle control libraries. Controlled in software using GNURadio libraries, the radio nodes in our emulation

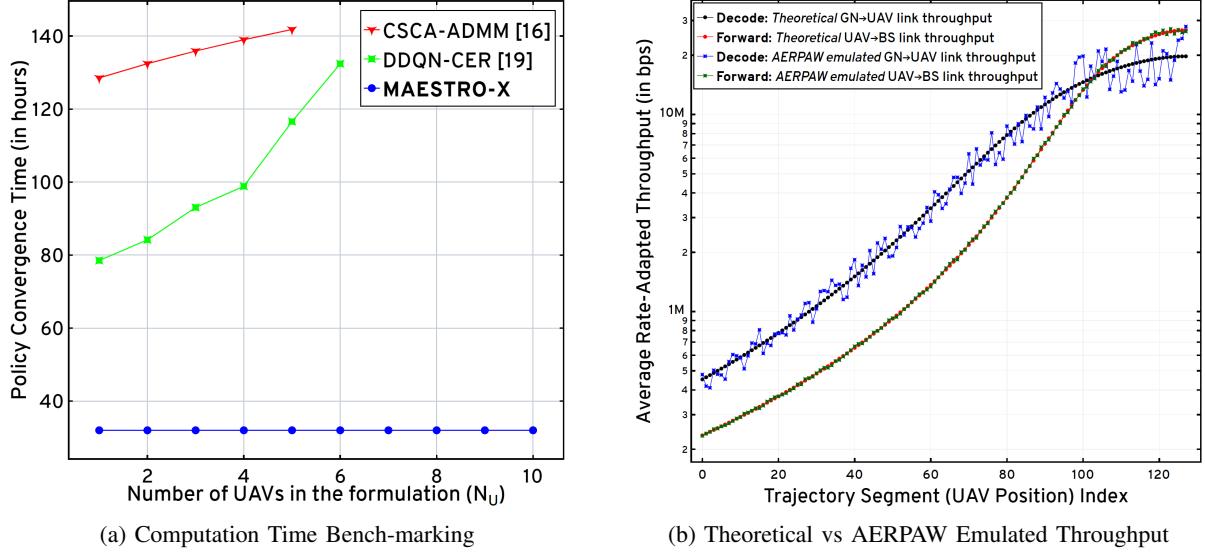


Fig. 7: (a) Computational bench-marking of MAESTRO-X against relevant frameworks in the state-of-the-art; (b) Comparisons between the theoretical rate-adapted throughput achieved in our simulations and that obtained during the emulated execution of the optimal trajectory for  $L=10$  Mb and  $P_{avg}=1.2$  kW on NSF AERPAW.

include the fixed BS (at the center of the cell), the fixed GN (at a designated position within the cell), and the UAV (portable node) flying at a fixed height of 200 m. To efficiently exploit A2G channel dynamics, at each of these radios, rate adaptation centers around an adaptive modulation and coding strategy with the instantaneous SNR fed back to determine the best modulation scheme and code rate for the transmitted symbols. We emulate the optimal service trajectory of the UAV shown in Fig. 5b, as it traverses its decode and forward paths, orchestrated from within MAESTRO-X via asynchronous remote calls using the platform-provided vehicle control libraries over the MAVLink interface: considering the same channel and mobility parameters, Fig. 7b depicts the comparison between the throughput achieved by our setup in this emulation environment against that achieved in our simulations, as the UAV traverses the  $M$ -segments ( $M=256$ ) of the optimal trajectory to serve the request. Here, we observe that the emulated throughput closely matches the theoretical one described in (4), which was derived under the idealistic assumptions of capacity achieving codes and throughput maximizing rate.

## VII. CONCLUSIONS

In this paper, we propose MAESTRO-X, a framework for the decentralized orchestration of a swarm of rotary-wing UAV relays in cellular networks, augmenting the coverage and service capabilities of a terrestrial BS. First, we specialize our system model to single-UAV deployments and design the optimal scheduling and trajectory optimization policy under an SMDP formulation (via value iteration and HCSO). Next, we extend this single-relay policy to distributed deployments of two or more UAVs by supplementing the policy with multi-UAV

coordination heuristics and M/G/ $x$  queuing management, and replicate this augmented policy across the swarm. Numerical evaluations demonstrate that MAESTRO-X delivers significant performance gains over BS-only and static UAV deployments; furthermore, MAESTRO-X outperforms popular Q-learning and successive convex approximation strategies in the existing literature. Finally, we demonstrate the integration and implementation viability of MAESTRO-X through emulations on the NSF AERPAW platform.

## REFERENCES

- [1] B. Keshavamurthy and N. Michelusi, “Multiscale Adaptive Scheduling and Path-Planning for Power-Constrained UAV-Relays via SMDPs,” in *ASILOMAR 2022 - Asilomar Conference on Signals, Systems, and Computers*, 2022.
- [2] B. Keshavamurthy, “MAESTRO-X: Multiscale Adaptive Energy-conscious Scheduling and TRajecotry Optimization (eXtended),” April 2022. [Online]. Available: <https://github.com/bharathkeshavamurthy/MAESTRO-X.git>
- [3] A. Fotouhi, H. Qiang, M. Ding *et al.*, “Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges,” *IEEE Communications Surveys Tutorials*, pp. 1–1, 2019.
- [4] M. Mozaffari, W. Saad, M. Bennis *et al.*, “A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems,” *IEEE Communications Surveys Tutorials*, pp. 1–1, 2019.
- [5] Y. Zeng, R. Zhang, and T. J. Lim, “Wireless communications with unmanned aerial vehicles: opportunities and challenges,” *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, May 2016.
- [6] Q. Wu, L. Liu, and R. Zhang, “Fundamental Trade-offs in Communication and Trajectory Design for UAV-Enabled Wireless Network,” *IEEE Wireless Communications*, vol. 26, pp. 36–44, 02 2019.
- [7] B. Keshavamurthy and N. Michelusi, “Learning-based Spectrum Sensing and Access in Cognitive Radios via Approximate POMDPs,” *IEEE Transactions on Cognitive Communications and Networking*, 2021.
- [8] D. Alvear, “Inside Verizon | Up to Speed | Drones, robots and the power of 5G,” July 2021.
- [9] S. Guillette, “Verizon News Archives | Robots, drones and sensors are changing the way we farm,” March 2019.
- [10] Y. Zeng, J. Xu, and R. Zhang, “Energy Minimization for Wireless Communication With Rotary-Wing UAV,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2329–2345, April 2019.
- [11] M. A. Abd-Elmagid and H. S. Dhillon, “Average Peak Age-of-Information Minimization in UAV-Assisted IoT Networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 2003–2008, 2019.
- [12] X. Hu, K.-K. Wong, K. Yang *et al.*, “UAV-Assisted Relaying and Edge Computing: Scheduling and Trajectory Optimization,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 10, pp. 4738–4752, 2019.
- [13] J. Chen and D. Gesbert, “Optimal positioning of flying relays for wireless networks: A LOS map approach,” in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.
- [14] C. You and R. Zhang, “3D Trajectory Optimization in Rician Fading for UAV-Enabled Data Harvesting,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 3192–3207, 2019.
- [15] R. K. Patra and P. Muthuchidambaranathan, “Optimisation of Spectrum and Energy Efficiency in UAV-Enabled Mobile Relaying Using Bisection and PSO Method,” in *2018 3rd International Conference for Convergence in Technology (I2CT)*, 2018, pp. 1–7.
- [16] Q. Hu, Y. Cai, A. Liu *et al.*, “Low-Complexity Joint Resource Allocation and Trajectory Design for UAV-Aided Relay Networks With the Segmented Ray-Tracing Channel Model,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 9, pp. 6179–6195, 2020.

- [17] A. E. A. A. Abdulla, Z. M. Fadlullah, H. Nishiyama *et al.*, “Toward Fair Maximization of Energy Efficiency in Multiple UAS-Aided Networks: A Game-Theoretic Methodology,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 1, pp. 305–316, Jan 2015.
- [18] Y. Li and L. Cai, “UAV-Assisted Dynamic Coverage in a Heterogeneous Cellular System,” *IEEE Network*, vol. 31, no. 4, pp. 56–61, July 2017.
- [19] H. Bayerlein, M. Theile, M. Caccamo *et al.*, “Multi-UAV Path Planning for Wireless Data Harvesting With Deep Reinforcement Learning,” *IEEE Open Journal of the Communications Society*, vol. 2, p. 1171–1187, 2021.
- [20] L. Wang, K. Wang, C. Pan *et al.*, “Multi-Agent Deep Reinforcement Learning-Based Trajectory Planning for Multi-UAV Assisted Mobile Edge Computing,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 73–84, 2021.
- [21] A. M. Koushik, F. Hu, and S. Kumar, “Deep  $Q$ -Learning-Based Node Positioning for Throughput-Optimal Communications in Dynamic UAV Swarm Network,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 554–566, 2019.
- [22] Q. Zhang, M. Mozaffari, W. Saad *et al.*, “Machine Learning for Predictive On-Demand Deployment of UAVs for Wireless Communications,” in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6.
- [23] M. Clerc, *Particle Swarm Optimization*, ser. ISTE. London ; Newport Beach: ISTE, 2006.
- [24] R. Cheng and Y. Jin, “A Competitive Swarm Optimizer for Large Scale Optimization,” *IEEE Transactions on Cybernetics*, vol. 45, no. 2, pp. 191–204, 2015.
- [25] Q. Wu, Y. Zeng, and R. Zhang, “Joint Trajectory and Communication Design for Multi-UAV Enabled Wireless Networks,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2109–2121, 2018.
- [26] M. Mozaffari, W. Saad, M. Bennis *et al.*, “Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage,” *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.
- [27] J. Hu, H. Zhang, and L. Song, “Reinforcement Learning for Decentralized Trajectory Design in Cellular UAV Networks With Sense-and-Send Protocol,” *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6177–6189, 2019.
- [28] H. Shakhreh, A. Khreichah, A. Alsarhan *et al.*, “Efficient 3D placement of a UAV using particle swarm optimization,” in *2017 8th International Conference on Information and Communication Systems (ICICS)*, 2017, pp. 258–263.
- [29] Z. Yuheng, Z. Liyan, and L. Chunpeng, “3-D Deployment Optimization of UAVs Based on Particle Swarm Algorithm,” in *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, 2019, pp. 954–957.
- [30] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal LAP Altitude for Maximum Coverage,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.
- [31] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge: Cambridge University Press, 2005.
- [32] R. Essaadali and A. Kouki, “A new simple Unmanned Aerial Vehicle doppler effect RF reducing technique,” in *MILCOM 2016 - 2016 IEEE Military Communications Conference*, 2016, pp. 1179–1183.
- [33] J. D. C. Little and S. Graves, *Little’s Law*, 07 2008.
- [34] S. Boyd, L. Xiao, and A. Mutapcic, “Subgradient methods,” *Lecture notes, Stanford University, Autumn Quarter*, vol. 2004, pp. 2004–2005, 2003.
- [35] Y. Zeng and R. Zhang, “Energy-Efficient UAV Communication With Trajectory Optimization,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, June 2017.
- [36] P. Yang, K. Tang, and X. Yao, “Turning High-Dimensional Optimization Into Computationally Expensive Optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 143–156, 2018.
- [37] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal LAP Altitude for Maximum Coverage,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.