IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

Boosting UAVs Live Uplink Streaming by Video Stabilization

ELEONORA DI SALVO[1], (Student, IEEE), AZEDDINE BEGHDADI[2], (Senior, IEEE), TIZIANA CATTAI[1], (IEEE), FRANCESCA CUOMO[1], (Senior, IEEE), and STEFANIA COLONNESE[1], (Senior, IEEE)

[1]Department of Information Engineering, Electronics and Telecommunications, Sapienza University of Rome,Rome, Italy (e-mail: name.surname@uniroma1.it)
[2]Institut Galilée, Sorbonne Paris Nord, Paris, France (e-mail: azeddine.beghdadi@univ-paris13.fr)

Corresponding author: Stefania Colonnese (e-mail: stefania.colonnese@uniroma1.it).

This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART"

ABSTRACT The high-impact scenario of UAV live uplink streaming is gaining significant interest in diverse applications, such as ambient monitoring, disaster rescue, and smart surveillance. This paper addresses the problem of uplink streaming by a fleet of camera-equipped UAVs, with one UAV acting as the sink, collecting and transmitting videos from the others. We demonstrate that performing video stabilization at the source UAVs or the sink enhances video quality and reduces required communication throughput, leading to bandwidth savings. We analyze the UAV live uplink streaming architecture to identify the most effective stabilization point within the network in a distributed manner. Using a reinforcement learning framework, we develop a method to dynamically optimize the stabilization gain-cost trade-off, pinpointing the optimal node for stabilization tasks. Through targeted numerical simulations under different system conditions we identify when and where stabilization should be applied to maximize efficiency. Our results show that video stabilization improves system performance in terms of media quality, battery life, and bandwidth usage.

INDEX TERMS stabilization,5G and 6G networks, autonomous systems, extended reality, quality metric.

I. INTRODUCTION

Uplink streaming in Unmanned Aerial Vehicles (UAVs) networks plays a crucial role in several cutting edge applications, such as ambience patrolling, advanced surveillance, visual localization and mapping, disaster recovery area searches, to cite a few. Fig.1 illustrates the architecture for live uplink streaming for UAV fleets, reflecting the 3GPP Framework for Live uplink streaming (FLUS) architecture [1]. Fig.1 shows the FLUS architecture elements, namely: i)source(s) UAV, i.e. camera equipped UAVs capturing video or volumetric data, and ii)sink UAV (optional), i.e. a UAV gathering and possibly processing the data from the sources. UAV live uplink streaming presents challenging constraints of low latency and high data rate typical of live video streaming, which can be satisfied in 5G and beyond networks. In the UAV uplink streaming scenario, UAV or vehicle mounted devices capture motion-intensive footage, such as conventional or immersive (e.g., 360 degree) video and extended reality media (e.g., point clouds) affected by shaking or global motion effects. The motion of the UAV camera impacts the perceptual quality of the acquired video, and processing techniques designed to reduce the apparent video motion, also known as video stabilization techniques [2], can be applied. In [2] the authors extensively review the literature on the topic and identify

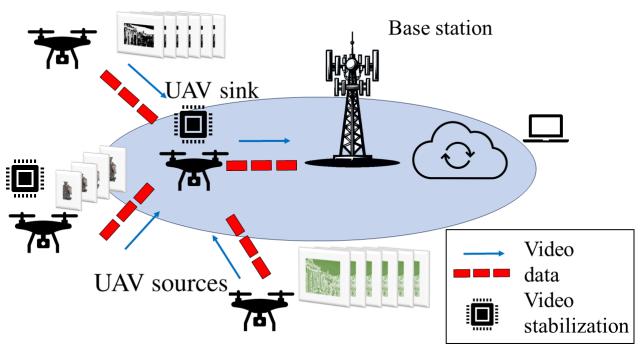

FIGURE 1. UAV uplink streaming scenario: source UAVs are equipped with cameras and transmitters, while the sink UAV, equipped with both receivers and transmitters, acts as a relay to the base station. All UAVs have limited battery life, operating under strict energy and bandwidth constraints. Video stabilization can reduce the system cost of uplink streaming.

open challenges, such as developing low computational cost methods and creating novel video datasets possibly annotated with subjective quality evaluation information. These datasets enable further research on the estimation of video stabilization quality using deep learning [3], [4], paving the way to online, automatic video quality assessment.

The literature overlooks the impact of video stabilization on communication resource savings, especially in resource-constrained UAV live streaming. This paper fills this gap by addressing video stabilization within the UAV uplink streaming framework.

The UAV can act as a video "source" or "sink" in the communication pipeline. The source UAV captures video frames, encodes the raw unstabilized video, and forwards it to the next communication element, such as a base station or an intermediate sink UAV. It can optionally stabilize the video before encoding and transmission, possibly collecting sensor data crucial for stabilization. The UAV sink collects videos from multiple UAVs and can relay them to a base station. If the video hasn't been stabilized at the source, the sink can decode, stabilize, and re-encode it for transmission. The sink can also provide feedback to the source for real-time adjustments, such as modifying capture settings. In a heterogeneous architecture, the UAV sink could be replaced by a sink mounted on an autonomous or human-operated vehicle.

This paper demonstrates that video stabilization reduces the bitrate of encoded video and analyzes its impact on the UAV fleet's bandwidth and energy resources. Using a reinforcement learning approach, we investigate the communication, computing, signaling costs and benefits of video stabilization at the source or sink UAV. After identifying protocol architectures for implementing video stabilization, we show how reinforcement learning can optimize resources for overall gains in stabilization and transmission.

To summarize, this paper brings the following novel contributions:

- We outline the architecture of UAV live uplink streaming, identifying key system parameters affecting performance.
- We assess the impact of video stabilization by comparing Rate-Distortion performance of stabilized versus shaky videos, highlighting potential bandwidth savings.
- We define a reinforcement learning approach for flexible, distributed optimization of computing and communication resources between source and sink UAVs, identifying the optimal element within the UAV fleet for stabilization.
- We provide numerical simulations that highlight the system conditions under which stabilization can be optimally performed.

This paper is structured as follows: Section II reviews relevant literature. Section III details the live UAV uplink streaming scenario. Section IV presents our novel strategy for integrating video stabilization into UAV operations by framing it as a Markov Decision Process (MDP), introducing Q-learning, and discussing different architectures. Section V addresses system-related issues, such as signaling overhead. Section VI assesses bandwidth savings from video stabilization, presents the benefits across various scenarios, and discusses the optimal location for stabilization (source vs. sink). Finally, Section VII concludes the paper and proposes directions for future research.

II. RELATED WORK

The boost in video acquisition by mobile devices has spurred video stabilization research, leading to various innovative approaches aimed at improving video quality by reducing unwanted shakiness. The work in [5] presents a probabilistic approach to minimize optical flows between shared regions among different reconstructed frames. In [6], recovery of affine transformation of the camera plan is achieved by a network trained on synthetic data. The algorithm in [7] combines deep learning with frame interpolation to prevent stabilization side effects such as blur. The deep online video stabilization method in [8] predicts steady frames with minimal latency, emphasizing the separation of motion estimation from camera path smoothing to reduce wobbling and distortion. The authors in [9] propose to adaptively adjust meshes to handle parallax and discontinuous depth variations in various scene complexities. [10] addresses the specific challenges of stabilizing selfie videos by focusing on both foreground and background motion, using a stabilization network to infer displacements and generate stabilized frames. Overall, stabilization of conventional video is an open problem [2]. Novel media kinds, such as 360 degree video or lidar data, are expected to enable novel services, e.g. visual simultaneous localization and mapping. These media pose further challenges for the stabilization algorithms.

For immersive video, after the seminal paper [11], several methods have been developed; see [12] for a survey. For lidar volumetric data as point clouds, stabilization poses extraordinary challenges since different temporal snapshots typically lack of vertices correspondences [13], and the literature mainly addresses small misalignment recovery (registration) rather than capturing and compensating for global motion [9]. These studies demonstrate interest in addressing processing complexities, but the literature overlooks the integration of video stabilization into the wide range of mobile video services provided by next generation networks, including those by UAV fleets.

Regarding uplink streaming in UAV networks, recent research has focused on enhancing communication capabilities, resource allocation, and addressing challenges related to channel modeling and interference. The work in [14] addresses resource allocation and power control within cellular networks supporting UAV communications, suitably accounting for inter-cell interference. The work [15] presents a prototype for real-time video streaming and UAV control, emphasizing the use of LTE networks for downlink and uplink transmissions. In [16], the application of cell-free massive MIMO technology in UAV networks, particularly for ultra-reliable low-latency communications is explored within a

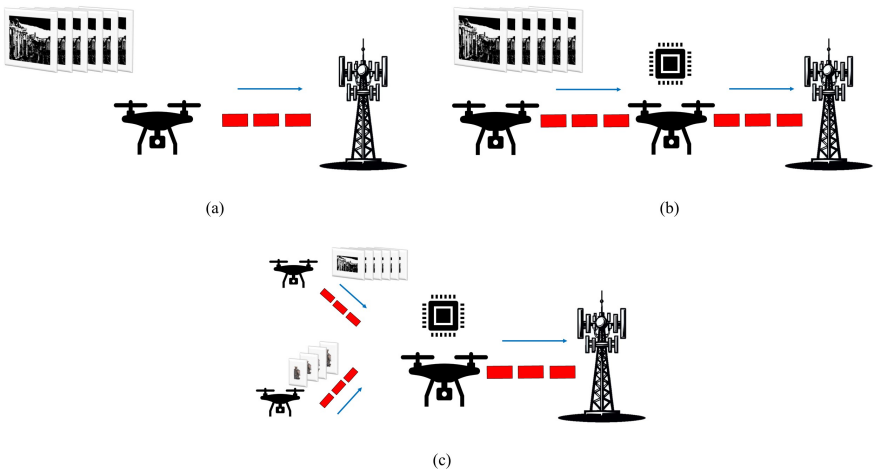

FIGURE 2. Different communication architectures for uplink streaming: (a) single-hop, representing a direct access by the source UAV to the base station; (b) two-hops, where a single UAV source is connected to a UAV sink, acting as a relay towards the base station; (c) two-hops, where the UAV sink gathers data from multiple UAVs and relays it towards the base station.

smart factory environment. Focusing on security, [17] maximizes effective secrecy throughput in uplink UAV networks under limited feedback channels. The work in [18] examines the impact of UAV trajectories and altitude on uplink non-orthogonal multiple access in cellular networks, whereas [19] proposes user-centric and UAV-centric strategies for enhancing UAV communications in a multi-cell setup. Regarding video streaming, [20] addresses power efficient UAV placement and resource allocation for adaptive video streaming, while [21] addresses the optimal movements and transmission powers for UAV fleets in real-time, maximizing the video rate, smoothness and low latency. An energy-related focus is provided in [22]. The work in [23] optimizes the positioning of observational and relay UAVs and allocates communication resources for uplink streaming in power safety scenarios. The interest in UAV-aided streaming is also demonstrated by various surveys, such as [24]–[26].

These studies advance understanding of uplink streaming in UAV networks, addressing interference management, real-time control, security, and radio access efficiency. However, the literature lacks focus on the novel and critical issue of efficient video stabilization within resource-constrained UAV fleets. The significant system advantages of UAV video stabilization will be explained next.

III. UAV UPLINK STREAMING ARCHITECTURE

In the UAV live uplink streaming scenario, one or more UAVs equipped with cameras (UAV sources) are deployed to cover an area capturing media as video or volumetric data [27]. They can either be directly connected to a base station or maintain a connection to a UAV sink equpped with an access point, which is connected to the base station. The UAV sink gathers the data, and transmits it to a receiving antenna. The different communication architectures are shown in Fig.2, illustrating the case of connection to the base station (a) through a direct link, (b) through a dedicated UAV sink, and (c) through a UAV sink shared by multiple UAV sources.

As the UAVs sources collect video footage, their aerial movements induce shakiness in the captured video. To address this, videos may undergo a stabilization process directly at the source UAV to improve quality before they are encoded and sent to the sink. The transmission rate of video from each UAV varies, with average value Θ_n[Mbps] for the n-th UAV, depending on whether the footage is stabilized prior to transmission.

In a UAV live uplink streaming scenario, each source UAV possesses a unique spectral efficiency, denoted as η_n (measured in bits per second per Hertz, bps/Hz), towards the sink [28]. This efficiency is influenced by the characteristics of the communication channel, notably including the UAV's distance from the sink. Consequently, the physical bandwidth required to upload the video from each UAV, represented as B_n, can be calculated using the formula $B_n = \Theta_n/\eta_n$, where B_n is expressed in Megahertz (MHz). In addition to communication capabilities, each UAV is also defined by its battery state, quantified as its residual battery charge, C_n. The sink collects the data from the source UAV: if the video footage has not been stabilized at the source - for instance because the source was not equipped with a stabilization algorithm- the sink has the option to decode, stabilize, and re-encode the video data. The sink then forwards the video to the connected base station at a transmission rate denoted by $\Theta_n^{(s)}$ (Mbps), which is less than or equal to Θ_n.

The bandwidth B_{sink} used by the sink for the transmission to the base station is computed as $B_{\text{sink}} = \Theta_n^{(s)}/\eta_{\text{sink}}$ [MHz], being η_{sink} [bps/Hz] the spectral efficiency of the channel between the sink and the base station. Additionally, the sink itself is characterized by its own battery charge C_{sink}.

In summary, the interaction between the source UAV and sink UAV in video stabilization involves a frame processing

Main notation	
Video parameters	
Θ_n, $\Theta_n^{(s)}$	Average throughput [Mbps] of the original and stabilized video originated by the n-th UAV.
τ	Video chunk duration [s].
γ_n, $\gamma_n^{(s)}$,	Size [Mbits] of the original and stabilized video chunk generated every τ s at the n-th UAV.
System parameters	
η_n, η_{sink}	Channel spectral efficiency [bps/Hz] of the n-th UAV-to-sink channel and the sink-to-base station channel.
$B_n, \tilde{B}_n, B_{\text{sink}}, \tilde{B}_{\text{sink}}$	Bandwidth required and allocated to upload the video from the n-th UAV (to the sink), and from the sink (to the base station).
C_n, C_{sink}	Residual battery charge at the n-th UAV and at the sink.
$\kappa_n^{(stab)}$, $\kappa_n^{(enc)}$, $\kappa_n^{(tx)}$	Energy cost [J/bit] for encoding, stabilization and transmission at the UAV.
$\kappa_{\text{sink}}^{(stab)}$, $\kappa_{\text{sink}}^{(enc)}$, $\kappa_{\text{sink}}^{(tx)}$	Energy cost [J/bit] for encoding, stabilization and transmission at the sink.
ρ_s, ρ_{BS}	Traffic load at the sink and the base station, statistically affecting the video packet delay.
MOSV, MOSS	Subjective quality metric (Mean Opinion Score, MOS) of the video quality (MOSV) and of its stabilization level (MOSS).
ΔR, ΔPSNR	Bjontegard metric representing the rate saving and the PSNR improvement achieved by video stabilization.
Performance metrics	
f_E, f_B, f_D,	Cost associated with the $\mathcal{S}_i \rightarrow \mathcal{S}_j$ state transition.
ΔE, ΔB (%)	Relative bandwidth and energy saving achieved by video stabilization.
$\varphi(U)$, $\varphi(S_{UAV})$,$\varphi(S_{Sink})$	Relative frequency φ of the optimal path estimated by the RL algorithm for the Unstabilized, UAV and Sink Stabilized video.

TABLE 1. Main notation for video, system and performance parameters.

pipeline. The source UAV captures or reads a video frame, optionally stabilizes it to correct motion artifacts, encodes the frame, and sends it to the sink UAV or the base station. The sink UAV, if present, relays the video to the base station. If stabilization wasn't applied at the source, the sink UAV can apply it before forwarding to the base station. This setup, although simplified, encapsulates the main system parameters affecting the quality of live UAV video streaming. It considers various critical factors such as mobility, video stabilization, data transmission efficiency, and the operational longevity of the UAVs. It may be used to describe different alternative architectures as shown in Fig.2(a)-(c).

IV. UAV VIDEO STABILIZATION AS A REINFORCEMENT LEARNING PROBLEM

In this section, we investigate resource savings using video stabilization in UAV uplink streaming by formulating a Markov Decision Process (MDP) and building a reinforcement learning model for video stabilization in a UAV fleet; the adopted notation is reported in 1. We focus on the architecture in Fig.2(b), where both the source and sink are in the video processing pipeline, and discuss modifications for the schemes in Fig.2(a), (c).

Considering the architecture in Fig.2(b), data from the source UAV travels through the sink UAV to the base station. We introduce an optimal strategy for the source UAV to autonomously decide whether to perform video stabilization or offload it to the sink UAV. This decision is based on minimal parameters and the UAV's current state. We formulate this optimization problem as an MDP and solve it using Q-learning. Reinforcement learning allows dynamic and continuous improvement of the optimization strategy online and supports broader applications, such as adjusting video bitrate or selecting transmitting UAVs.

A. UAV STABILIZATION AS A MARKOV DECISION PROCESS (MDP)

A Markov Decision Process (MDP) is a system composed of a sequence of states where transitions from one state to another are associated with a reward, dependent only on the current state. Among all the possible sequences of states, there is one that maximizes the overall reward. The MDP can thus be represented as a graph, where the optimal sequence of states corresponds to a path in the graph.

We first describe the MDP underlying our framework, highlighting the system parameters affecting the MDP transition rewards. Then, we solve the problem using the Q-learning algorithm, which simulates an agent that visits the graph multiple times (epochs) and explores different paths, estimating the cumulative path rewards [28]. After a predefined number of epochs, the agent identifies the optimal path. Applying this computation method by the UAV leads to identifying the best solution for video stabilization.

The literature [28] often addresses learning UAV path planning and area coverage, associating the UAV state with variables such as the position, height, and velocity. Herein, we focus on a novel aspect: finding the optimal strategy (allocation of computation and communication resources) for the video stabilization and transmission by developing a model that is novel compared to the state-of-the-art. Our model addresses the temporal scale of video packets, which typically spans a few seconds. In contrast, existing algorithms that optimize UAV actions, such as hovering, being idle, or charging, operate on larger temporal scales. These longer-term algorithms can integrate the proposed algorithm within one or more of their states, such as during the UAV's hovering or charging phases, optimizing video packet transmission within the broader context of UAV operations.

We specify the MDP model to perform the stabilization in the UAV fleet, either at the source UAVs or at the sink. The source UAV, equipped with a camera, captures the video,

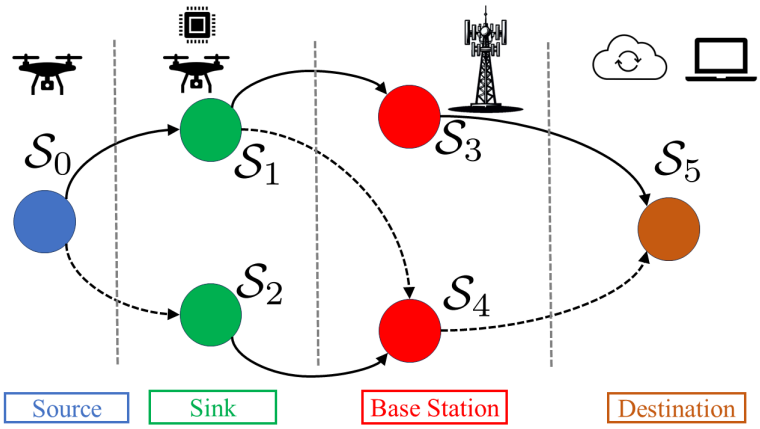

FIGURE 3. **Markov decision process (MDP) model representing the optimal video stabilization strategy as an optimal path on a graph. The graph model is the base for the proposed reinforcement learning approach, where the optimal path is iteratively learned.**

encodes and transmits it. The acquired video can be stabilized before encoding and transmission. It is worth noting that video stabilization has a direct positive impact on coding performance. Indeed, the smoothing effect of the motion vector field resulting from the video stabilization process enhances the efficiency of the encoder's motion estimation and compensation block. As described in the previous section, the video stabilization reduces the amount of motion in the video, potentially reducing the throughput (bps) needed to transmit the video at a given target visual quality. The UAV source transmits the video to the UAV sink; the bandwidth occupied for the transmission depends on the video features, possible stabilization and encoding efficiency, as well as by the spectral efficiency η_n characterizing the channel between the source and sink UAVs. The videos collected at the sink are forwarded to the network through the base station. The sink can either transmit the video from the UAV source as is or decode, stabilize, re-encode and forward it to the base station. We assume the sink can stabilize the video only if it has not been previously stabilized at the source. In principle. The bit-rate reduction achievable by video stabilization ρ_{sink} differs from that achievable at the source. Finally, the transmission towards the base station is realized over a channel with spectral efficiency η_{sink}.

We introduce the stabilization strategy for the scenario of Fig.2(b), where the video can be stabilized either at the source UAV or at the sink UAV. We disregard the case of stabilizing the video twice, as the system cost would outweigh the marginal benefits of a second stabilization stage, as demonstrated in the numerical results. In SubsectionIV-C, we will discuss the cases in Fig.2(a) and (c). The stabilization strategy for Fig.2(b) is represented by the MDP indicated in Fig.3. The state $\mathcal{S}_0$ represents the source UAV; states $\mathcal{S}_1$ and $\mathcal{S}_2$ refer to the availability of the encoded or the stabilized and encoded video at the UAV sink; states $\mathcal{S}_3$, $\mathcal{S}_4$ refer to the availability of the encoded video, either stabilized or not at the base station. The state $\mathcal{S}_5$ represents the end destination. It is assumed that the link between the base station and the final destination, e.g. a central command room, is wired, and the further in-network impact of the rate saving due to the stabilization is negligible.

An online, dynamic approach to the problem is obtained by using a reinforcement learning algorithm [28], where an agent travels from $\mathcal{S}_1$, to $\mathcal{S}_6$, visiting the states along a path. The links stemming from each state represent the set of possible actions available in each state. This is sketched in Fig.3, where the continuous and dashed lines represent the actions *encoding* and *stabilization and encoding*, respectively. The decision on whether and where to stabilize the video is made based on optimizing the path from $\mathcal{S}_1$ to $\mathcal{S}_6$ based on relevant system optimization variables, as described in detail below.

B. MDP SOLUTION BY Q-LEARNING ALGORITHM

As shown in [29], deep learning can be applied to UAVs using a variety of algorithms, leveraging reinforcement learning, deep reinforcement learning, transformer models, and general deep neural networks. Training large models significantly enhances performance, using [30] gradient-based machine learning algorithms or pretrained neural networks for faster inference [31]. Although attention-based algorithms have proven to be highly performant [32], [34] or fast [33] in different applications, deep learning, reinforcement learning and their developments are typically preferred for managing missions with multiple cooperative UAVs [35], using transformer-based deep reinforcement learning for efficient routing, even in a multi-agent framework for better scalability [36].

For simplicity and without loss of generality, we focus on reinforcement learning implemented using the Q-learning algorithm. This allows us to highlight all the benefits and trade-offs associated with UAV onboard video stabilization and provides a lower bound of performance achievable by adopting advanced algorithms.

Let $\mathcal{E}$ denote the set of edges of the MDP graph in Fig.3.

The objective is to compute the system reward for each state-action pair, represented by the Q-matrix $Q : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$,

Action	Energy cost f_E	Bandwidth cost f_B	QoE cost f_δ
$\mathcal{S}_0 \to \mathcal{S}_1$	$\left(\kappa^{(enc)}\gamma_n + \kappa^{(tx)}\gamma_n\right)/C_n$	$\eta_n\gamma_n, E\{\gamma_n\} = \Theta_n \cdot \tau$	$\max\left(\tilde{B}_n/B_n - 1, 0\right) \Leftrightarrow \rho_{sink}$
$\mathcal{S}_0 \to \mathcal{S}_2$	$\left(\kappa^{(stab)}\gamma_n + \kappa^{(enc)}\gamma_n^{(s)} + \kappa^{(tx)}\gamma_n^{(s)}\right)/C_n$	$\eta_n\gamma_n, E\{\gamma_n\} = \Theta_n^{(s)} \cdot \tau$	$\max\left(\tilde{B}_n/B_n - 1, 0\right) \Leftrightarrow \rho_{sink}$
$\mathcal{S}_1 \to \mathcal{S}_3$	$\kappa_{\text{sink}}^{(tx)}\gamma_n/C_{\text{sink}}$	$\eta_{\text{sink}}\gamma_n, E\{\gamma_n\} = \Theta_n \cdot \tau$	$\max\left(\tilde{B}_{\text{sink}}/B_{\text{sink}} - 1, 0\right) \Leftrightarrow \rho_{BS}$
$\mathcal{S}_1 \to \mathcal{S}_4$	$\left(\kappa_{\text{sink}}^{(stab)}\gamma_n^{(s)} + \kappa_{\text{sink}}^{(dec)}\gamma_n + \kappa_{\text{sink}}^{(enc)}\gamma_n^{(s)} + \kappa_{\text{sink}}^{(tx)}\gamma_n^{(s)}\right)/C_{\text{sink}}$	$\eta_{\text{sink}}\gamma_n, E\{\gamma_n\} = \Theta_n^{(s)} \cdot \tau$	$\max\left(\tilde{B}_{\text{sink}}/B_{\text{sink}} - 1, 0\right) \Leftrightarrow \rho_{BS}$
$\mathcal{S}_2 \to \mathcal{S}_4$	$\kappa_{\text{sink}}^{(tx)}\gamma_n^{(s)}/C_{\text{sink}}$	$\eta_{\text{sink}}\gamma_n, E\{\gamma_n\} = \Theta_n^{(s)} \cdot \tau$	$\max\left(\tilde{B}_{\text{sink}}/B_{\text{sink}} - 1, 0\right), \Leftrightarrow \rho_{BS}$

TABLE 2. Cost functions f_E, f_B, f_δ associated with actions from state $\mathcal{S}_i$ to $\mathcal{S}_j$ of the graph in Fig.3. The cost functions depend on several system parameters concerning the sources and sink UAVs energy costs, battery values, and spectral efficiency, the overall sink and base station load.

where $\mathcal{S} = \{\mathcal{S}_0, \cdots \mathcal{S}_5\}$, $\mathcal{A} \leftrightarrow \mathcal{E}$ are the available actions corresponding to the MDP edges. The algorithm refines the Q matrix over numerous iterations (epochs), using the estimate from each epoch as a starting point for the next.

During each epoch, the reinforcement learning agent moves from $\mathcal{S}_0$, to $\mathcal{S}_5$. At each step k, given the current state s_k, the agent selects an action $a_k \in \mathcal{A}$ and transitions to the next state s_{k+1}. The action a_k in state s_k yields a reward $\mathcal{R}_k = \mathcal{R}(s_k, a_k)$, updating the Q-matrix. Specifically, let $Q_{k-1}^{(i)}$ be the Q matrix estimated at $k-1$ step in the i-th epoch. At step k, the $Q_{k-1}^{(i)}$ matrix is updated based on i) the reward $\mathcal{R}_k$ of action a_k associated with the transition from state s_k to state s_{k+1}, and ii) the potential reward achievable from the next state s_{k+1}, i.e. $\max_{a_{k+1}} Q_{k-1}^{(i)}(s_{k+1}, a_{k+1})$. In formulas,

$$Q_k^{(i)}(s_k, a_k) = (1-\alpha_i) \cdot Q_{k-1}^{(i)}(s_k, a_k) + \alpha_k \left[R_k + \lambda \max_{a_{k+1}} Q_{k-1}^{(i)}(s_{k+1}, a_{k+1})\right], \quad (1)$$

for $k = 1, \cdots 5$, $i = 0, \cdots N_e$, where α_i in $[0, 1]$ is the step size or learning rate, and λ in $[0, 1]$ is the discount factor that balances the importance of earlier versus later rewards.

During each epoch, the Q-learning algorithm employs an ϵ-greedy policy to balance exploration and exploitation: with probability $1 - \epsilon$, it chooses the action a_k that maximizes the expected reward for the current state s_k, and with probability ϵ, it selects an action at random. The advantage of this approach is that it prevents premature convergence to a local optimum.

Upon convergence, the refined Q matrix determines the optimal sequence of states, indicating the best action from any given state to maximize cumulative reward.

We apply the Q-learning algorithm within the uplink streaming framework. Let n denote the index of the n-th video packet transmitted during the uplink streaming session. The Q-matrix is iteratively estimated by computing the QoE reward for each state-action pair (i.e. each transition in Fig.3) as a function of system parameters.

We focus on the uplink transmission of a video packet, or chunk, of duration τ and random size γ_n depending on the average video encoded data rate Θ. The rate Θ varies based on whether the video is stabilized or not, i.e. $\Theta \in \left\{\Theta_n^{(s)}, \Theta_n\right\}$. Typically, γ_n follows a heavy-tailed distribution such as a Gamma distribution $p_\Gamma(\gamma; \Theta)$ [37] of expected value $E\{\gamma_n\} = \Theta \cdot \tau$. Let β_n denote the bit-rate reduction achievable by video stabilization at the n-th UAV sink, i.e. $\beta_n = \Theta_n^{(s)}/\Theta_n$. The bandwidth B_n requested by the n-th source UAV for chunk transmission depends on the spectral efficiency η_n of the channel towards the sink, i.e. $B_n = \gamma_n \cdot \tau$. The actually allocated bandwidth $\tilde{B}_n$ depends on the total load of the UAV sources covered by the UAV sink, so that $\tilde{B}_n \leq B_n$. When $\tilde{B}_n < B_n$, a transmission delay occurs. For the common case of HTTP-based streaming [27] the delay can be quantified as

$$\delta_n = \tau \cdot \max\left(B_n/\tilde{B}_n - 1, 0\right).$$

Generally, δ_n statistically depends on the sink traffic load ρ_s. Transmitting the chunk requires energy that sums up the energy spent on the stages performed at the UAV, possibly including the stabilization $E_n^{(stab)} = \kappa^{(stab)}\gamma_n$, coding $E_n^{(enc)} = \kappa^{(enc)}\gamma_n$, and transmission $E_n^{(tx)} = \kappa^{(tx)}\gamma_n$ energy costs.

Finally, the allocated bandwidth from the base station to the sink, namely $\tilde{B}_{\text{sink}}$, depends on the total load of the base station. When $\tilde{B}_{\text{sink}}$ is less than the overall bandwidth requested by the sink B_{sink}, a delay $\delta_{\text{sink}} = \tau \cdot \max\left(\tilde{B}_{\text{sink}}/B_{\text{sink}} - 1, 0\right)$ occurs even the sink and the base station. The delay probability distribution depends on the base station traffic load level ρ_{BS}. The energy spent at the sink includes the stabilization $E_{\text{sink}}^{(stab)} = \kappa_{\text{sink}}^{(stab)}\gamma_n$, coding $E_{\text{sink}}^{(enc)} = \kappa_{\text{sink}}^{(enc)}\gamma_n$, decoding $E_{\text{sink}}^{(dec)} = \kappa_{\text{sink}}^{(dec)}\gamma_n$, and transmission $E_{\text{sink}}^{(tx)} = \kappa_{\text{sink}}^{(tx)}\gamma_n$ energy costs.

With these settings, we are can define the rewards associated with each action in the above-defined state-action space. Specifically, at each action, we associate the reward in terms of energy and bandwidth savings and quality of experience maintenance.

Let us denote by f_E, f_B the cost in terms of energy and bandwidth spent for the chunk transmission and by f_δ the cost in terms of transmission delay. In its general form, we define the reward as the opposite of three cost functions

$$\mathcal{R} = -\left(f_E + f_B + f_\delta\right) \quad (2)$$

where the actual costs f_E, f_B, f_δ vary across links and can be set depending on several system parameters.

In the following, we will restrain ourselves to consider the impact of the relevant system parameters:

- the processing, coding and transmission energy at the n-th source $E_n^{(stab)}, E_n^{(enc)}, E_n^{(tx)}$, and at the sink $E_{\text{sink}}^{(stab)}, E_{\text{sink}}^{(enc)}, E_{\text{sink}}^{(tx)}$;

- the encoding rate of the original video Θ_n and of the stabilized video $\Theta^{(s)}$;
- the battery charge at the n-th UAV source and at the UAV sink;
- the sink and base station load condition, as summarized by the parameters ρ_{sink} and ρ_{BS}, which ultimately influences the observed ratios $\tilde{B}_n/B_n$, $n=0,\cdots N-1$.

These settings, detailed in Tab.2, allow using the Q-learning algorithm to compute the Q-matrix associated to the model in Fig.3 from a set of parameters related to the uplink streaming session. The Q-learning algorithm can be applied offline and online, either in a centralized or distributed way, provided that a few parameters are either available or signaled, as discussed in Sec.V. Once the Q matrix is estimated, the optimal path in Fig.3 and the corresponding stabilization strategy is selected.

C. REMARKS

We discuss here some key aspects of the possible alternative architectures in Fig.2. In the model in Fig.2(a), the only decision strategy at the source UAV is whether to stabilize the video or not. Therefore, the reward is simplified, and we straightforwardly write it as

$$\mathcal{R} = \left(\kappa_n^{(enc)}\gamma_n + \kappa_n^{(tx)}\gamma_n\right)/C_n + \eta_n\gamma_n + \max\left(\tilde{B}_n/B_n - 1, 0\right) \tag{3}$$

or as

$$\mathcal{R}^{(s)} = \left(\kappa_n^{(stab)}\gamma_n + \kappa_n^{(enc)}\gamma_n^{(s)} + \kappa_n^{(tx)}\gamma_n^{(s)}\right)/C_n + \eta_n\gamma_n^{(s)} + \max\left(\tilde{B}_n/B\right. \tag{4}$$

where the delay statistics depends in both cases on ρ_{BS}. We recognize that in this case the reward depends on the trade-off between the stabilization energy cost and the available bandwidth.

The model in Fig.2(c) involves different UAVs, and the optimal strategy is ideally determined jointly for all the UAVs using a multi-agent approach. In this case, the reward component related to the QoE depends on the bandwidth requested by each and every UAV. Exchanging this information would imply a large signaling overhead. Therefore, we limited ourselves to a suboptimal distributed solution, requiring very few parameters to be known at the UAV source to select the better strategy. The bandwidth requests from all the users could be accounted for by a key parameter representing the overall load of the UAV sink. This load parameter can be accounted for in the above described model by introducing a weight w_B for the bandwidth reward component f_B; for loads close to the maximum UAV sink bandwidth, the sink UAV would send the signaling parameter to the competing UAV sources, which would then react to improve the reward component associated with the bandwidth saving.

Finally, all the above described strategies can be integrated into more general path planning strategies by suitably incorporating the stabilization-related reward components to the existing ones.

V. SYSTEM RELATED ISSUES

In this Section, we tackle some relevant issues concerning the implementation of the method in a real system.

a) *Path planning.* Most of the UAV-related literature uses reinforcement learning for path planning purposes [28]. The decision process presented here is suitable for adoption within a more general path planning strategy. Specifically, the MDP in Fig.3 operates on a time scale of the order of seconds and can be integrated as a chain of sub-states within a higher layer path planning model, where the states typically corresponds to larger temporal scales of UAV dynamics. This is exemplified in Fig.4, where the higher level (macro) UAV states refer to dynamics such as hovering, moving or charging. Each macro state can be extended to enclose the current model as a set of sub-states. The reinforcement algorithm is modified by extending both the state by adding the relevant parameters and the reward to include the component related to the stabilization algorithm.

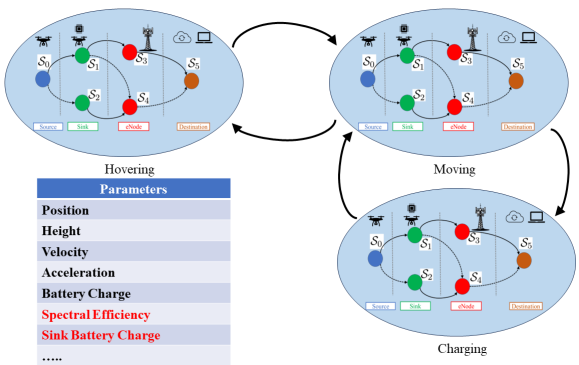

FIGURE 4. Integration of the proposed learning strategy within UAV path planning: the video stabilization strategy operates on a shorter temporal scale, with actions occurring over intervals of a few seconds. Thus, stabilization can be incorporated as a series of substates within higher-level states. The state parameters (black) are integrated with all relevant parameters (red).

b) *Parameters' signaling.* The optimal stabilization strategy is selected by reinforcement learning, typically performed offline before the session begins and updated online throughout the session [27]. To perform the online reinforcement learning task, some parameters need to be exchanged. Fig.5 shows how the signaling takes place during the session: some parameters are estimated locally, whereas others are received from the UAV sink. The parameters that are constant during the session, such as the energy consumption per processed/encoded bit, are sent in the initialization phase (solid line); the parameters that vary during the session, such as the sink battery charge state, are updated throughout the session (dashed line).

c) *Computational complexity.* The proposed strategy involves various steps, including video data processing, encoding, and learning. Real-time solutions are available for encoding [38], [39] and for digital video stabilization [40], [41], with latency reduced to one or less frame

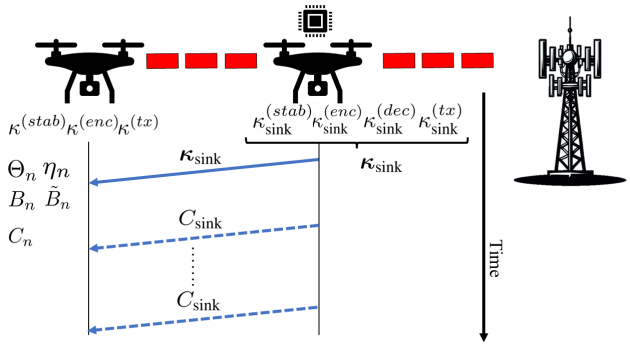

FIGURE 5. Parameter signaling for online distributed strategy learning: each UAV source applies online the reinforcement learning strategy, based on: i) the parameters estimated locally and ii) the parameters received from the UAV sink either in the link initialization phase (solid line) and during the session (dashed line).

[42], possibly using gyroscope measurements [43]. The learning is realized in two steps: an offline training stage, where the system parameters are set up, and an online stage, where the strategy is updated. In case of overload at the UAV sink, the signaling parameters outlined earlier can be used to discourage processing at the UAV sink (e.g., in case of multiple UAVs and heavy sink load) while promoting video stabilization at the UAV source.

d) *Scalability.* Multi-UAV systems are known to present scalability issues in path planning and task allocation. A comprehensive review is found in [44], highlighting the scalability challenges related to different classes of tasks, such as coverage, adversarial search/game, computational offloading, communication, and target-driven navigation. The case of uplink streaming may fall within tasks of coverage or target-driven navigation. Besides the challenges highlighted in [44], mainly due to the exponential growth of the state and action spaces with the number of UAVs, the approach presented here faces an additional bottleneck at the UAV sink. Realizing the video stabilization at each UAV source, i.e., in a distributed manner, releases resources at the UAV sink for processing video data that could not be processed otherwise, such as in cases of severe battery discharge at the originating UAV. For scalability purposes, when the number of UAVs increases, more than one UAV can be elected to act as a data processing sink. This architecture paves the way for a further hierarchical organization of the fleet, based on the amount of the data to be processed, the path of the UAVs and their system resources.

VI. NUMERICAL EXPERIMENTS

In this section, we present the numerical simulation results for unmanned aerial vehicles (UAVs) providing uplink streaming services by application of video stabilization before coding. The architecture in Fig.2(b) allows us to draw some general trends on the system functioning, and it will be analyzed in detail in the following.

Firstly, we show that integrating video stabilization significantly enhances video compression efficiency. Then, we analyze the considerable system resource savings achieved by applying stabilization techniques in UAV uplink streaming. Specifically, we address the impact of video stabilization on the encoded video bitrate and the associated quality in Subsec.VI-A. In Subsec.VI-B we investigate the impact in a UAV system. With reference to the architecture in Fig.3, we show the performance of the optimal video stabilization strategy as a function of the UAV and sink system state and connectivity condition, both through MDP optimization and reinforcement learning.

A. VIDEO STABILIZATION IMPACT ON THROUGHPUT: A RATE-DISTORTION ANALYSIS

This subsection delves into the influence of video stabilization on the video's Rate-Distortion (R-D) trade-off during the encoding stage, providing a detailed examination through experiments. It presents empirical results focusing on two critical metrics: Peak Signal to Noise Ratio (PSNR) and Mean Opinion Score (MOS), which serve as indicators of objective video quality and subjective viewer satisfaction, respectively [45]. The analysis demonstrates that video stabilization not only enhances these quality metrics but also contributes to significant bandwidth savings. By stabilizing the video, it becomes possible to transmit high-quality video content more efficiently, reducing the amount of data that needs to be sent over the network.

Our experiments, detailed below, show that the stabilized videos, being smoother in time, exhibit an encoding rate for a given objective quality level that is systematically and significantly below the rate of the original shaky videos. This is due to the typically global nature of the shaking motion, which is not effectively compressed by conventional hybrid transform-based encoders.

We conduct encoding experiments on 320 videos [46], comprising 64 unstabilized and their differently stabilized counterparts, using the HEVC-based libx265 codec. We evaluate video compression quality by calculating the PSNR, i.e., the ratio between the squared peak signal value (255^2) and the mean square of the encoding error. Quantization parameters were adjusted between 20 to 50, in steps of 5 to compare encoding performance. Fig.6 shows the PSNR as a function of the encoding bitrate for both the original and the stabilized videos. In Tab.3 we report the results obtained between original and stabilized videos encoded using one of the latest ITU-T video encoder, namely H.265 [47] at different quantization levels.

On these results, we compute the Bjontegard metrics [48] representing the bitrate saving $\boldsymbol{\Delta}$R and the quality improvement $\boldsymbol{\Delta}$PSNR of the stabilized videos with respect to the original video, for the sequences in Fig.6.

So far, we have computed the quality of the stabilized video using an objective quality metric, namely the PSNR. We now assess the quality of the stabilized video using the subjective

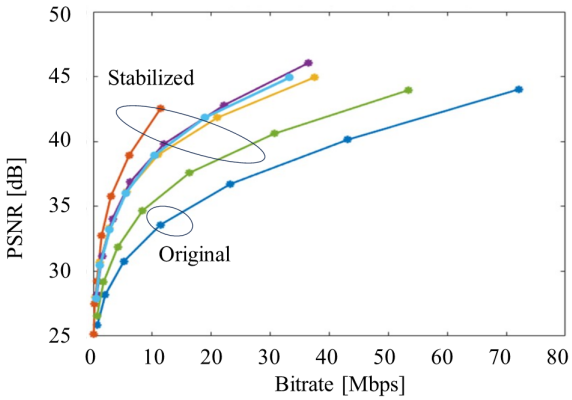

FIGURE 6. **Visual quality, measured by PSNR, versus bitrate achieved by encoding the original video (blue curve) and the videos stabilized with different state-of-the-art algorithms [46]. For a given visual quality (PSNR), the stabilized video bitrates can reduce up to 20% of the original video bitrate.**

TABLE 3. **Bjontegard ´R and ´PSNR metrics for the stabilization algorithms (i) Adobe After Effects (Ae) warp stabilizer, (ii) Google Photos application, (iii) VideoProc Converter, (iv) VirtualDub Deshaker and (v) vReveal described in [46].**

	Ae	GoogleP	VPC	VDub	VRev
ΔPSNR [dB]	-5.82	-3.30	-4.27	-2.76	-0.90
ΔRate [kbps]	902.3	471.3	589.9	400.6	256.2

quality metric adopted in [46]. Specifically, we considered the dataset in [46], which reports the subjective evaluation of stabilization quality and video quality, both measured in terms of Mean Opinion Score (MOS), as an average of the subjective results obtained from a measurement campaign. The linear fit between the perceived stabilization quality (MOSS) and the perceived video quality (MOSV) measured over a few stabilized versions of the video on a single video is shown in Fig.7. The linear fit analysis has been repeated over all the videos in the dataset in [46]. The fitting straight lines are illustrated in Fig.8(a), and the cumulative distribution of the line slopes is shown in Fig.8(c). In most cases (highlighted in

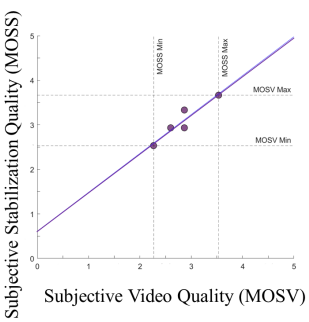

FIGURE 7. **Subjective quality (Mean Opinion Score) evaluating the stabilization level (MOSS) and the video quality (MOSV) for a single video stabilized by different algorithms (purple points); a linear fitting curve of MOSS versus MOSV is shown (dotted blue line).**

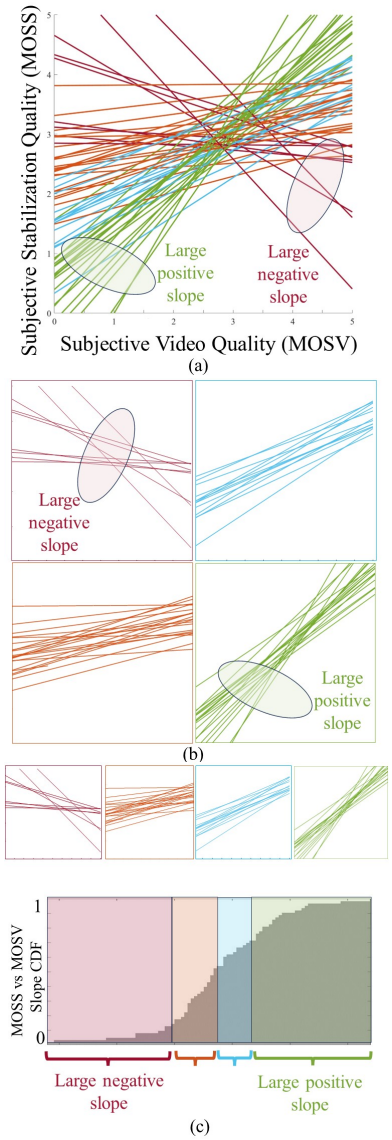

FIGURE 8. **Linear fitting curves of the Mean Opinion Score evaluating the stabilization level (MOSS) and the video quality (MOSV) for the entire video dataset in [46]: (a) linear fitting lines,(b) selected set of lines with different slopes, and (c) cumulative distribution of the lines' slopes. A positive slope indicates that video quality increases with stabilization level, whereas a negative slope indicates that stabilization may decrease quality. We recognize that instances with large positive slopes are significantly more frequent than those with large negative slopes: in most cases (green, cyan, orange lines in (a) and green, cyan, orange areas in (c)), the MOSS increases with MOSV; only in a few cases (red lines in (a) and red region in (c)), increasing the level of stabilization decreases the quality due to processing artifacts such as video smoothing or cropping.**

green or cyan) the perceived video quality increases with the level of stabilization; still, there are a few cases (highlighted in orange) where the phenomenon is less appreciated and even a small fraction of cases (highlighted in red) where the contrary happens. This result is consistent with other findings in the literature [49], which report that a varying percentage of users may prefer a shaky video over an unsatisfying stabilized version. The quality is perceived through both local and global features [50], associated with the object contours and recognizability. Stabilization may improve global stability but can also introduce local artifacts such as smoothing or boundary cropping. Therefore, there is a need for an objective QoE metric to assess the actual advantages of the stabilization in terms of QoE quality.

The above findings are pivotal for the next section, which is devoted to the reinforcement learning-driven allocation of computing and communication resources in the UAV fleet.

We highlight the following: i) the potential bandwidth saving achieved by video stabilization, and ii) the optimal RL strategy for different system conditions, such as the sink and network load and UAV and sink charge.

B. REINFORCEMENT LEARNING FOR NETWORK ASSISTED VIDEO STABILIZATION

Previous experiments on unstabilized and differently stabilized videos [46], both encoded using the H.265 HEVC codec [47], assessed the encoding rate gain achievable by video stabilization. We now show how to select the stabilization strategy within a live uplink streaming service using reinforcement learning. We will refer to the case of a single-drone, multi-hop case represented in Fig.2(b), and then generalize the conclusion for the cases in Fig.2(a) and (c). Specifically, we show the impact of system state, such as battery charges or network load, on the best selected strategy. We then consider an agent associated with one UAV, identifying the best strategy given a small number of system parameters, so that the learning can be implemented either at the UAV or at the sink, provided that minimal state information is signaled.

The reinforcement learning algorithm is based on $N_{epoch} = 100$, with $\epsilon = 0.75$, discount factor $\lambda = 0.5$ and learning rate $\alpha = 0.7$. Each epoch corresponds to the transmission of one $\tau = 1$s long video chunk from the UAV to the core network across one of the paths in Fig.3.

Fig.9 illustrates the chunk sizes, γ_n, for both unstabilized and stabilized videos, which are generated using a heavy-tailed Gamma distribution with a mean of $\Theta_n \cdot \tau = 6$ Mbits. For each video chunk, the size of the stabilized version is calculated assuming a reduction factor of $1/\beta$, where $\beta = 3$. This significant reduction in chunk size highlights distinct traffic characteristics between the unstabilized and stabilized videos, which are expected to result in substantial bandwidth savings during the transmission of the stabilized video.

We now evaluate the resource savings achievable through video stabilization by analyzing the optimal path within the MDP, as depicted in the graph in Fig.3. The energy related factors may vary significantly over different devices and conditions [52], but here we are only interested in their relative weight. To compute the rewards, we set the factors to the same value, apart for the decoding which is set to 20% of the other costs [51].

In Fig.10 we observe the percentage gains in energy consumption and bandwidth usage for stabilized video at both the UAV and the sink across different values of the ratio $\beta = \Theta_n/\Theta_n^{(s)}$, specifically $\beta = 2, 3, 4, 5$. Stabilization at the UAV results in gains in both energy and bandwidth. In contrast, while stabilization at the sink also yields bandwidth savings, it incurs a higher energy cost (negative savings), due to the transcoding processes (decoding, stabilization, encoding) that occur at the sink. Nevertheless, depending on system parameters such as low UAV battery charge, stabilization at the sink may still be a viable option.

We now explore the implementation of the decision strategy using a reinforcement learning algorithm, specifically Q-learning. Fig.11 illustrates the costs associated with energy, bandwidth, and delay, presented from top to bottom for two scenarios: (a) low UAV battery charge and (b) low sink battery charge. These cost metrics consider the battery state and guide the reinforcement learning algorithm towards a sequence of states that best align with the available device resources.

We now examine the achievability of the optimal path using a learning algorithm. Fig.12 displays the relative frequency of the optimal path as estimated by the Q-learning algorithm over 100 epochs for various video chunk size distributions: (a) deterministic, (b) uniformly distributed within a $\pm 10\%$ interval around the mean, (c) uniformly distributed within a $\pm 20\%$ interval around the mean, and (d) gamma distributed with a standard deviation of $\pm 50\%$ of the mean. These results indicate that the convergence of the Q-learning algorithm is influenced by the variability of video traffic, characterized by settings (100 epochs, $\epsilon = 0.9$, $\gamma = 0.5$, $\alpha = 0.5$). Specifically, the algorithm achieves higher accuracy when the video chunk size distribution exhibits low variance. Conversely, high variance in the distribution impedes convergence. In an uplink streaming scenario, it is anticipated that the online encoder will implement rate control to minimize rate fluctuations at its output buffer, thereby reducing jitter. This stabilization is advantageous for the learning algorithm's estimation of the optimal path.

C. DISCUSSION

The above findings underscore the dual benefits of video stabilization in UAV-operated uplink streaming services: economizing bandwidth and conserving energy, which are critical factors in enhancing the operational efficiency and sustainability of UAV communications networks.

We can draw a few key takeaways:

1. Significant bandwidth and energy gains are observed when video stabilization occurs at the device acquiring the video; stabilizing the video at an intermediate stage requires additional energy to decode the video bitstream before stabilization. The UAV's video stabilization yields bandwidth and

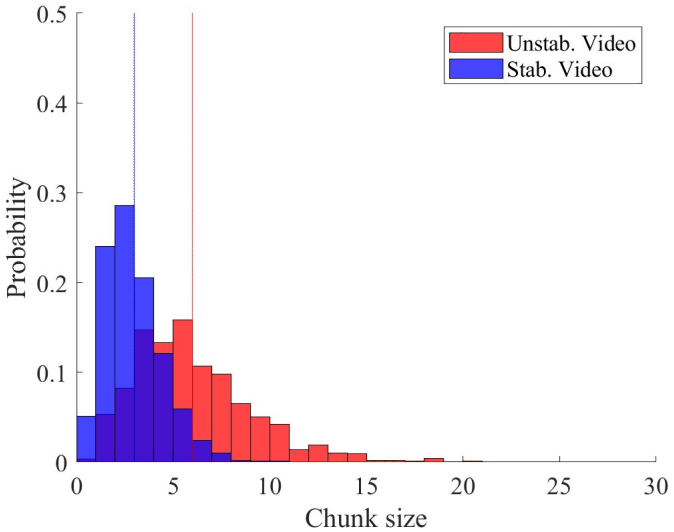

FIGURE 9. **Video packet (chunk) size γ_n for the unstabilized and stabilized videos are simulated using a heavy-tailed Gamma distribution [37] with mean $\hat{} \cdot \tau$ = 6Mbits; for each chunk, the stabilized video chunk size is obtained assuming a size reduction of a factor $1/\beta$ with $\beta = 3$.**

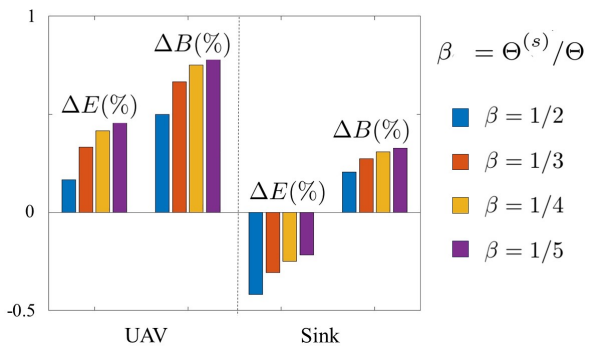

FIGURE 10. **Percentage gain of stabilized video in terms of energy consumption and bandwidth occupation, at the UAV and at the sink for different values of the rate saving $\beta_n = \tilde{}^{(s)}_n / \hat{}_n$, namely $\beta = 1/2,\ 1/3,\ 1/4,\ 1/5$. The UAV stabilization yields energy and bandwidth gains; the sink stabilization yields bandwidth gains but is paid in terms of a larger bandwidth expense (negative savings). This is due to the transcoding (decoding, stabilization, encoding) that takes place at the sink when stabilization is performed. Still, stabilization at the sink may still be adopted depending on system parameters, e.g. when the UAV has low battery charge.**

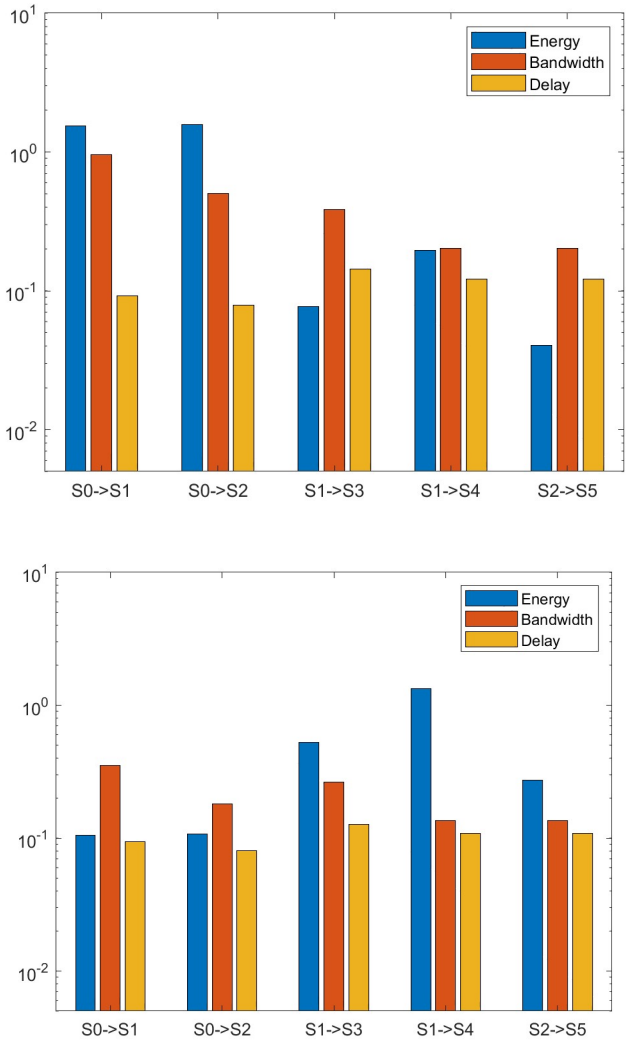

FIGURE 11. **Cost of the energy, bandwidth and delay components, top to bottom: (a) case of low UAV battery charge and (b) case of low sink battery charge. The cost terms account for the battery state and are used to drive the reinforcement learning solution towards the state sequence best matching the actual device resources.**

energy saving across various levels of rate reduction due to the smoothing of the video footage.

2. These results allow us to infer the best strategies for the setup in Fig.2(a), where the sole UAV source is responsible for video stabilization. Provided that the UAV has the required hardware capability, video stabilization shall always be applied for an effective trade-off energy versus bandwidth consumption, as previous results show that it brings both energy and bandwidth reduction.

3. With reference to the case in Fig.2(c), previous results suggest that there is an additional energy cost when stabilization is performed at the sink, since a transcoding operation is needed. Hence, video stabilization should be performed at the sink only when strictly required to overcome the lack of bandwidth at the base station, such as in case of high traffic from the sink gathering multiple UAV sources. In any case, the bandwidth gain will be come at the cost of medium to high energy consumption at the UAV sink and should be restrained in time.

The UAVs and sink can dynamically select the video stabilization strategy based on partial state information, such as knowledge of their battery charge state. The strategy selection by reinforcement learning is viable for integration within a more general reinforcement learning strategy, such as identifying the UAV trajectories. However, the intrinsic variability of the UAV uplink streaming traffic packets hinders the convergence of the reinforcement learning algorithm. This

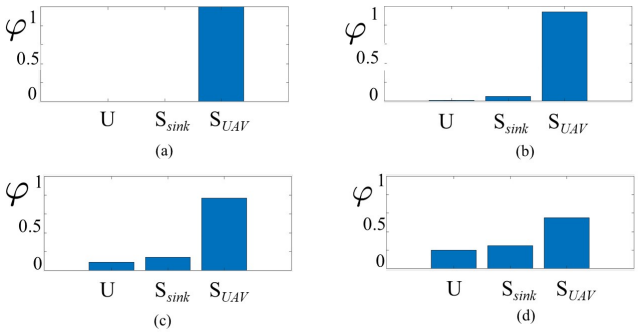

FIGURE 12. Relative frequency φ of the optimal path estimated by the Q-learning algorithm for the Unstabilized, UAV and Sink Stabilized video, and for different distributions of the video chunk size γ_n: (a) deterministic, (b) uniformly distributed in a $\pm$10% interval around the mean, (c) uniformly distributed in a $\pm$20% interval around the mean, (d) gamma distributed with standard deviation equal to $\pm$50% of the mean. The convergence of the Q-learning algorithm depends on the video traffic variability (100 epochs,$\epsilon = 0.9, \gamma = 0.5, \alpha = 0.5$).

can be mitigated by a simplified version of the algorithm; for example, stabilization may be performed at the UAV unless its battery charge falls below a threshold, in which case stabilization is performed at the sink.

D. REAL WORLD LIMITATIONS

The above results may be hindered by implementation issues related to real world limitations. From a service architecture point of view, the main limitation is the increased latency associated with video stabilization, which can be about 40 ms [42]. Therefore, if the video is delivered within a real-time service, such as patrolling or fleet control, the improvement in throughput/quality assured by the stabilization is traded-off with an increased delay in video packet delivery at the final destination. Additionally, processing may accelerate the battery discharge of the involved UAVs, reducing the operational duration of the UAV service. From a communication channel perspective, video stabilization allows the system to cope with reduced channel capacity thanks to the bandwidth saving achieved by the stabilization.

A remark is due regarding the capacities of the relay node. Having the video stabilized at the UAV sources reduces incoming and outgoing throughput at the UAV-sink, which is beneficial both in terms of saved bandwidth and energy. However, offloading the stabilization to the UAV-sink by multiple connected drones may rapidly saturate the drone computational capabilities. Therefore, in a real-world system, the election of a UAV as sink should be updated based on the UAV current battery state. More generally, we observe that, in the case of a heterogeneous fleet, the sink can also be an autonomous, non flying vehicle with different battery charge and specifics than UAV. This allows the processing/relay task to scale up exploiting communication and computing resources on a dedicated vehicle. With these premises, media acquisition and processing could scale up to 3D scene representation using multiple cameras or LIDAR, and video stabilization can be extended to preprocessing different media kinds. The study of these extended application is left for future research.

VII. CONCLUSION AND FUTURE WORK

This paper explores the impact of video stabilization on UAV-based uplink video streaming. It examines a network of camera-equipped UAVs deployed to hover over a specific area while connected to a UAV sink that collects and transmits data to a receiving antenna. After assessing improvements in video compression efficiency yielded by video stabilization, we modeled the selection of the optimal video stabilization strategy—choosing no stabilization, stabilization at the sink, or stabilization at the source UAV—as a Markov Decision Process, which we solved using reinforcement learning.

Our numerical results demonstrate how UAVs and sinks can dynamically select video stabilization strategies through reinforcement learning. This approach can also be integrated into broader reinforcement learning challenges, such as UAV path planning. Numerical simulations indicate that significant bandwidth and energy savings are achieved when video stabilization is implemented at the source UAV. Alternatively, stabilizing at the sink may be preferable in scenarios of low UAV battery charge.

For future work, extending the benefits of video stabilization to more comprehensive video preprocessing tasks appears promising. In goal-oriented communication tasks, where UAV footage is utilized for specific knowledge extraction like moving object detection, preprocessing on the UAV side can substantially reduce system costs associated with uplink streaming challenges.

REFERENCES

[1] 3GPP TR 26.939, "Guidelines on the Framework for Live Uplink Streaming (FLUS),V17.0.0 (2021-09)

[2] Wang, Y., Huang, Q., Jiang, C., Liu, J., Shang, M. and Miao, Z., "Video stabilization: A comprehensive survey." Neurocomputing, 516, 2023, pp.205-230.

[3] Kou, T., Liu, X., Sun, W., Jia, J., Min, X., Zhai, G. and Liu, N., "Stablevqa: A deep no-reference quality assessment model for video stability", *Proceedings of the 31st ACM International Conference on Multimedia* (pp. 1066-1076), october 2023.

[4] E. M. Mahmoodpour, A. Amirany, M. H. Moaiyeri, and K. Jafari, "A Learning Based Contrast Specific no Reference Image Quality Assessment Algorithm," International Conference on Machine Vision and Image Processing (MVIP), 2022.

[5] Zhao, W., Li, X., Peng, Z., Luo, X., Ye, X., Lu, H. and Cao, Z., " Fast Full-frame Video Stabilization with Iterative Optimization", *Proceedings of the IEEE/CVF International Conference on Computer Vision* 2023 (pp. 23534-23544).

[6] Kerim, A., Ramos, W.L., Marcolino, L.S., Nascimento, E.R. and Jiang, R., " Leveraging synthetic data to learn video stabilization under adverse conditions," *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* 2024 (pp. 6931-6940).

[7] Choi, J. and Kweon, I.S., "Deep iterative frame interpolation for full-frame video stabilization," ACM Transactions on Graphics (TOG), 39(1), 2020, pp.1-9.

[8] Zhang, Z., Liu, Z., Tan, P., Zeng, B. and Liu, S., " Minimum latency deep online video stabilization." *Proceedings of the IEEE/CVF International Conference on Computer Vision* 2023 (pp. 23030-23039).

[9] Zhao, M. and Ling, Q., "Adaptively meshed video stabilization," IEEE Transactions on Circuits and Systems for Video Technology, 31(9), 2020, pp.3504-3517.

[10] Yu, J., Ramamoorthi, R., Cheng, K., Sarkis, M. and Bi, N., " Real-time selfie video stabilization"', *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* 2021 (pp. 12036-12044).

[11] Kopf, J., "360 video stabilization," ACM Transactions on Graphics (TOG), 35(6), 2016, pp.1-9.

[12] Fan, C.L., Lo, W.C., Pai, Y.T. and Hsu, C.H., "A survey on 360 video streaming: Acquisition, transmission, and display," Acm Computing Surveys (Csur), 52(4), 2019, pp.1-36.

[13] Muresan, M.P., Giosan, I. and Nedevschi, S., "Stabilization and validation of 3D object position using multimodal sensor fusion and semantic segmentation"', Sensors, 20(4), 2020, p.1110.

[14] Mei, W., Wu, Q. and Zhang, R., "Cellular-connected UAV: Uplink association, power control and interference coordination," IEEE Transactions on wireless communications, 18(11), 2019, pp.5380-5393.

[15] Zhou, H., Hu, F., Juras, M., Mehta, A.B. and Deng, Y., "Real-time video streaming and control of cellular-connected UAV system: Prototype and performance evaluation," IEEE Wireless Communications Letters, 10(8), 2021, pp.1657-1661.

[16] Peng, Q., Ren, H., Pan, C., Liu, N. and Elkashlan, M., "Resource allocation for uplink cell-free massive MIMO enabled URLLC in a smart factory," IEEE Transactions on Communications, 71(1), 2022, pp.553-568.

[17] Bastami, H., Behroozi, H., Moradikia, M., Abdelhadi, A., Ng, D.W.K. and Hanzo, L., "Large-Scale Rate-Splitting Multiple Access in Uplink UAV Networks: Effective Secrecy Throughput Maximization Under Limited Feedback Channel," IEEE Transactions on Vehicular Technology, 72(7), 2023.

[18] Senadhira, N., Durrani, S., Zhou, X., Yang, N. and Ding, M., " Uplink NOMA for cellular-connected UAV: Impact of UAV trajectories and altitude," IEEE Transactions on Communications, 68(8), 2020, pp.5242-5258.

[19] Hou, T., Liu, Y., Song, Z., Sun, X. and Chen, Y., " Exploiting NOMA for UAV communications in large-scale cellular networks," IEEE Transactions on Communications, 67(10), 2019, pp.6897-6911.

[20] Ahmed, Z., Ahmad, A., Altaf, M. and Khan, F.A., " Power efficient UAV placement and resource allocation for adaptive video streaming in wireless networks," Ad Hoc Networks, 150, 2023, p.103260.

[21] L. A. b. Burhanuddin, X. Liu, Y. Deng, U. Challita and A. Zahemszky, "QoE Optimization for Live Video Streaming in UAV-to-UAV Communications via Deep Reinforcement Learning," in IEEE Transactions on Vehicular Technology, vol. 71, no. 5, pp. 5358-5370, May 2022.

[22] Wu, K., Cao, X., Yang, P. and Yu, Z., "QoE-Driven Video Transmission: Energy-Efficient Multi-UAV Network Optimization," 11(1) IEEE Transactions on Network Science and Engineering 2024.

[23] Khan, N., Ahmad, A., Wakeel, A., Kaleem, Z., Rashid, B. and Khalid, W., " Efficient UAVs Deployment and Resource Allocation in UAV-Relay Assisted Public Safety Networks for Video Transmission," 12, IEEE Access 2024.

[24] Gu, X. and Zhang, G., " A survey on UAV-assisted wireless communications: Recent advances and future trends". 2023, Computer Communications.

[25] Nguyen, T.V., Nguyen, N.P., Kim, C. and Dao, N.N., " Intelligent aerial video streaming: Achievements and challenges, " Journal of Network and Computer Applications, 211, 2023 p.103564.

[26] Sharma, M.K., Liu, C.F., Farhat, I., Sehad, N., Hamidouche, W. and Debbah, M.," UAV Immersive Video Streaming: A Comprehensive Survey, Benchmarking, and Open Challenges, " arXiv preprint arXiv:2311.00082.

[27] Brunori, D., Colonnese, S., Cuomo, F., Flore, G. and Iocchi, L., " Delivering Resources for Augmented Reality by UAVs: a Reinforcement Learning Approach," Frontiers in Communications and Networks, 2, 2021, p.709265.

[28] Colonnese, S., Cuomo, F., Pagliari, G. and Chiaraviglio, L., " Q-SQUARE: A Q-learning approach to provide a QoE aware UAV flight path in cellular networks," Ad Hoc Networks, 91, p.101872.

[29] Puente-Castro, A., Rivero, D., Pazos, A. and Fernandez-Blanco, E., " A review of artificial intelligence applied to path planning in UAV swarms," Neural Computing and Applications, 34(1), 2022, pp.153-170.

[30] A. Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Le, Q. V., ... & Ng, A. Y. "Large scale distributed deep networks," In Advances in neural information processing systems 2012 (pp. 1223-1231).

[31] H. Imani, A. Amirany, T. El-Ghazawi " Mixture of Experts with Mixture of Precisions for Tuning Quality of Service," arXiv preprint arXiv:2407.14417, 2024.

[32] F. Chen, C., Zhang, M., Liu, Y., & Ma, S. " Neural attentional rating regression with review-level explanations," *Proceedings of the 2018 World Wide Web Conference* (pp. 1583-1592).

[33] H. Shafaghi, M. Kiani, A. Amirany, K. Jafari, M.H. Moaiyeri " A Fast and Light Fingerprint-Matching Model Based on Deep Learning Approaches," Journal Signal Processing Systems, vol. 95, no. 4, pp. 551-558, Apr 2023.

[34] C. Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. " BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer," *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* 2019 (pp. 1441-1450).

[35] Fuertes, D., del-Blanco, C.R., Jaureguizar, F., Navarro, J.J. and García, N. " Solving routing problems for multiple cooperative Unmanned Aerial Vehicles using Transformer networks, " Engineering Applications of Artificial Intelligence, 122, 2023, p.106085.

[36] Chen, D., Qi, Q., Fu, Q., Wang, J., Liao, J. and Han, Z., " Transformer-based reinforcement learning for scalable multi-uav area coverage", 25(8), IEEE Transactions on Intelligent Transportation Systems 2024.

[37] Colonnese, S., Frossard, P., Rinauro, S., Rossi, L. and Scarano, G., "Joint source and sending rate modeling in adaptive video streaming", Signal Processing: Image Communication, 28(5), pp.403-416.

[38] Hamidouche, W., Biatek, T., Abdoli, M., François, E., Pescador, F., Radosavljević, M., Hamidouche, W., Biatek, T., Abdoli, M., François, E., Pescador, F., Radosavljević, M., Menard, D. and Raulet, M., "Versatile video coding standard: A review from coding tools to consumers deployment," IEEE Consumer Electronics Magazine, 11(5), 2022, pp.10-24.

[39] W. Zhao, X. Li, Z. Peng, X. Luo, X. Ye, H. Lu, et al., "Fast Full-frame Video Stabilization with Iterative Optimization," *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 23534-23544) 2023.

[40] C. Jia and L. Brian, "Evans, Online motion smoothing for video stabilization via constrained multiple-model estimation," EURASIP J. Image Video Process., vol. 2017, no. 1, 2017.

[41] N. Joshi, W. Kienzle, M. Toelle, M. Uyttendaele, and F. Michael, "Cohen, Real-time hyperlapse creation via optimal frame selection," ACM Trans. Graph., vol. 34, no. 4, 2015.

[42] S. Liu, P. Tan, L. Yuan, J. Sun, and B. Zeng, "Meshflow: Minimum latency online video stabilization," in: European Conference on Computer Vision, Springer, 2016, pp. 800-815.

[43] C. Li, L. Song, S. Chen, R. Xie, and W. Zhang, "Deep Online Video Stabilization Using IMU Sensors," IEEE Trans. Multimed., vol. 25, pp. 2047–2060, 2023. DOI: 10.1109/TMM.2022.3142429

[44] Frattolillo, Francesco, Damiano Brunori, and Luca Iocchi "Scalable and cooperative deep reinforcement learning approaches for multi-UAV systems: A systematic review," Drones 7, no. 4 (2023): 236.

[45] I. Bezzine, Z. A. Khan, A. Beghdadi, N. Al-Maadeed, M. Kaaniche, S. Al-Maadeed, et al., " Video quality assessment dataset for smart public security systems," 2020 IEEE 23rd International Multitopic Conference (INMIC) (pp. 1-5).

[46] B. E. Dakkar, A. Bourki, A. Beghdadi, and F. A. Cheikh, "VStab-QuAD: A New Video-Stabilization Quality Assessment Database," 2023 11th European Workshop on Visual Information Processing (EUVIP) (pp. 1-6).

[47] International Telecommunication Union, "High Efficiency Video Coding," ITU-T Recommendation H.265, ISO/IEC 23008-2:2017, 2018.

[48] G. Bjontegaard, "Calculation of average PSNR differences between RD curves," ITU-T SG16/Q6 input document VCEG-M33, 2001.

[49] W. Guilluy, L. Oudre, and A. Beghdadi, "Video stabilization: Overview, challenges and perspectives," Signal Process. Image Commun., vol. 90, p. 116015, 2021.

[50] L. Wu, X. Zhang, H. Chen, D. Wang, and J. Deng, "VP-NIQE: An opinion-unaware visual perception natural image quality evaluator," Neurocomputing, vol. 463, pp. 17–28, 2021.

[51] M. Yan, C. A. Chan, A. F. Gygax, J. Yan, L. Campbell, A. Nirmalathas, et al., "Modeling the Total Energy Consumption of Mobile Network Services and Applications," Energies, vol. 12, no. 1, p. 184, 2019.

[52] P. K. D. Pramanik, N. Sinhababu, B. Mukherjee, S. Padmanaban, A. Maity, B. K. Upadhyaya, et al., "Power consumption analysis, measurement, management, and issues: A state-of-the-art review of smartphone battery and energy usage," IEEE Access, vol. 7,182113–182172, 2019.

[53] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Edge intelligence for energy-efficient computation offloading and resource allocation in 5G beyond," IEEE Trans. Vehicular Technol., vol. 69, no. 10, pp. 12175–12186, 2020.

ELEONORA DI SALVO Eleonora Di Salvo (Student Member, IEEE) received the BS degree in clinical engineering and the MS degree in biomedical engineering from Sapienza, University of Rome, Italy, in 2020 and 2023, respectively. She is currently working toward the PhD degree in Information and Communication Technology (ICT) at Sapienza, University of Rome. Her research interests focus on multimedia communications and extended reality services in next generation networks.

AZEDDINE BEGHDADI has been a full professor at the Sorbonne Paris Nord University since 2000. He received Diplôme d'Etudes Approfondies in Optics and Signal Processing from University Paris-Saclay in June 1983 and the PhD in Optics and Signal Processing from Sorbonne University Pierre and Marie in June 1986. He is the founding member of the Laboratory of Information Processing and Transmission (L2TI laboratory) and was its director from 2010 to 2016. Dr. Beghdadi is the founder and Steering Committee Chair of the European Workshop on Visual Information Processing (EUVIP). He has co-authored more than 350 peer-reviewed journal and conference papers including 6 best paper awards and has given over 30 invited and plenary talks. He supervised more than 40 PhD/Post-Doc and around 35 master students. His research interests include image/video quality enhancement and assessment, computational visual perception models for image/ video processing and scene understanding and medical imaging. Dr Beghdadi is senior associate editor of IEEE Transactions on Image processing, associate editor of IEEE Transactions on Multimedia, "Signal processing: Image Communication" and Journal, Elsevier, European journal on image and video processing, Springer Verlag. Dr Beghdadi is an elected member of IEEE-MMSP TC since 2021 and an elected member of EURASIP VIP-TAC (2021-2023). He is a senior member of IEEE, EURASIP member and IEEE-MMTC member.

TIZIANA CATTAI is Assistant Professor with the Department of Information Engineering, Electronics and Telecommunications (DIET), Sapienza University of Rome, Italy. She obtained her joint PhD from the Brain and Spine Institute at Sorbonne University, Paris, and Sapienza University of Rome with a thesis on brain connectivity networks to detect mental states. Prior to that, she completed a Master's degree in Biomedical Engineering and a Bachelor's degree in Clinical Engineering, both from the Sapienza University of Rome, graduating cum laude. Her research interests include signal processing, graph signal processing applied to IoT sensor networks, neuroscience and extended reality.

FRANCESCA CUOMO (Senior Member, IEEE) is Full Professor at Sapienza University of Rome teaching courses in Telecommunications, Network Infrastructures and Smart Environments. Prof. Cuomo has advised numerous master students in computer engineering and has been the advisor of 14 PhD students in Networking. She is the Chair of the Master Degree in Data Science in Sapienza.Her current research interests focus on Low Power Wide Area Networks and Internet of Things, 5G and Open Radio Access Networks, Smart Metering and Vehicular networks, Multimedia Networking. She participated in several national and international research projects, being Principal Investigator of many on them. From May 2022 she is part of the Scientific Committee of the Fondazione Ugo Bordoni and Technical Member of Namex (Italian IXP). Francesca Cuomo has authored over 185 peer-reviewed papers published in prominent international journals and conferences. Her Google Scholar h-index is 33 with >4700 citations. Relevant scientific international recognitions: 2 Best Paper Awards. She has been in the editorial board of Computer Networks (Elsevier), Ad-Hoc Networks (Elsevier), IEEE Transactions on Mobile Computing, Sensors (MDPI), Frontiers in Communications and Networks Journal. She has been the TPC co-chair of several editions of the ACM PE-WASUN workshop, TPC Co-Chair of ICCCN 2016, TPC Symposium Chair of IEEE WiMob 2017, General Co-Chair of the First Workshop on Sustainable Networking through Machine Learning and Internet of Things (SMILING), in conjunction with IEEE INFOCOM 2019; Workshop Co-Chair of AmI 2019: European Conference on Ambient Intelligence 2019. She will be track chair of IEEE WCNC 2025.

STEFANIA COLONNESE (Senior Member, IEEE) is Associate Professor with the Department of Information Engineering, Electronics and Telecommunications (DIET), Sapienza University of Rome, Italy. She has coauthored more than a hundred journal articles and conference papers, two book chapters, and several ISO MPEG-4 contributing documents. Her research interests include statistical signal processing, image deconvolution and restoration, and biomedical signal processing to video encoding, processing, and networking. She serves as Associate Editor for IEEE Transactions on Multimedia, devoted to the topics of multimedia broadcasting, standardization, and quality of experience.