

When UAVs Ride A Bus: Towards Energy-efficient City-scale Video Surveillance

Angelo Trotta*, Fabio D'Andreagiovanni[‡], Marco Di Felice*, Enrico Natalizio[‡], Kaushik Roy Chowdhury[†]

* Department of Computer Science and Engineering, University of Bologna, Italy

Email: {trotta, difelice}@cs.unibo.it

[‡] Sorbonne Universités, Université de Technologie de Compiègne, Heudiasyc, UMR CNRS 7253, France.

Email: {d.andreagiovanni, enrico.natalizio}@hds.utc.fr

[†] Department of Electrical and Computer Engineering, Northeastern University, Boston, USA

Email:krc@ece.neu.edu

Abstract—This paper proposes a network architecture and supporting optimization framework that allows Unmanned Aerial Vehicles (UAVs) to perform city-scale video monitoring of a set of Points of Interest (PoI). Our approach is systems-driven, relying on experimental studies to identify the permissible number of hops for multi-UAV video relaying in a noisy 3-D environment. Our architecture itself is innovative in the sense that it defines a mathematical framework for selecting the UAVs for periodic re-charging by landing on public transportation buses, and then ‘riding’ the bus to the successive chosen PoI. Specifically, we show that our UAV scheduler can be modeled as an instance of multi-commodity flow problems, and mathematically solved through Mixed Integer Linear Programming (MILP) techniques. Thus, our centralized formulation identifies the UAV, the next bus, and the next PoI, given the information about energy thresholds, the bus routes in the city and their next arrival times, to ensure persistent and reliable video coverage of all PoIs in the city. Finally, our work is validated via emulation of a city environment with live traffic updates from a real bus transportation network.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) are being increasingly deployed for next-generation surveillance of urban areas [2], driven by availability of off-the-shelf devices and reducing costs. High-resolution cameras, sensors and GPS can be mounted on top of low-cost quadcopters, guaranteeing better area monitoring than ground sensor networks [3] [5]. However, most readily deployable solutions today are based on a single UAV-ground station link, with limited flying time imposed by the on-board battery. This challenges the notion that a city-scale area can be effectively and persistently monitored by a UAV network under rigid performance bounds. Our approach addresses this important shortcoming through an innovative architecture that involves UAVs using public transportation for re-charging and traveling to the expected points of interest.

A. Challenges in city-scale monitoring

While major research efforts have focused on forming and coordinating UAV fleets, very few experimental studies provide a systems-driven and quantitative performance values of real-time video streaming involving multi-hop UAV communication [6] [7]. The urban channel introduces a harsh multipath reflection and scattering environment, which is worsened by

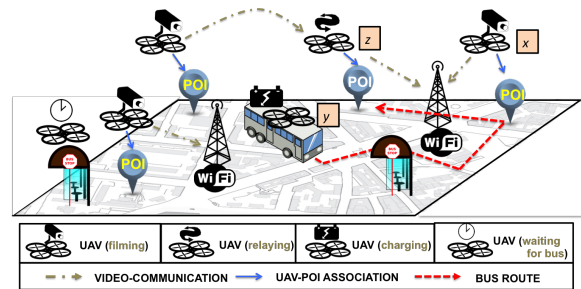


Fig. 1. Proposed architecture with UAVs filming city Point of Interests (POIs).

the continuous, albeit minor 3-D displacements of UAVs, even if they hover at a stationary point. Under such situations, we need to validate the number of intermediate hops that permit video reception, without unacceptable jitter and re-buffering. Furthermore, for persistent monitoring, the UAVs must rely on well positioned ground charging stations, with recent experimental results indicating the feasibility of this technology [8] [9]. One option for the recharging process is to use inductive coupling or simply *swap* batteries [10]. Accordingly, several recent studies have focused on scheduling algorithms to maximize the UAV service persistence over time and space, when they periodically visit static ground charging stations [4] [9] [11] [12]. In our vision of sustainable and cost-effective recharging solutions, we envision mobile charging stations that do not require specialized vehicles or robots. Instead, we assume that there is an underlying public transportation network of a city-run fleet of buses with established routes, where buses create the energy reserves during their regular motion. While this reduces the cost of operation of the UAVs, our proposed architecture complicates the problem of persistent coverage. Finally, whether at all sufficient energy can be transferred to the UAV in the short ‘riding’ time on a bus needs to be demonstrated; these practical challenges are spread across the topics of systems design, communications and network formation, and must be jointly tackled.

B. Why should UAVs ride a bus?

We propose a solution to a specific problem: how to continuously collect and communicate video streams from a large

number of Points of Interest (PoI) that are spread across a large urban city (see Figure 1). Our approach takes as inputs a set of PoI coordinates within an urban city, the set of bus routes, with the marked stops with the next arrival time of the bus at any given instant. This data is readily accessible using published API from most city-run transportation systems. We first create multihop forwarding paths, based on experimental studies of video quality degradation over multiple relaying UAVs. The UAVs themselves are assigned to distinct PoIs, till energy considerations and bus arrival schedules require them to switch to a charging mode. A given UAV x vacates its PoI, and lands on the chosen bus upon its arrival to a nearby stop. A different, charged UAV y , currently riding the bus, takes over its place for monitoring. The UAV x now rides the bus, also utilizing the time to re-charge using standard inductive coupling plates. It travels to the next PoI that is close to the bus route, and where a pre-existing UAV z is almost depleted of energy. The UAV swap occurs again with the charged UAV x resuming flight and video monitoring at the new PoI. This cycle continues as the bus travels along the city, continuously swapping UAVs at successive PoIs. Our work mathematically identifies the best time for a particular UAV to initiate the charging process, the PoI at which it should resume operations, and which bus route to select for maintaining persistent coverage, under constraints of acceptable video degradation. Thus, our framework defines the process of a UAV *riding* the bus, which serves the dual purpose of charging the UAV as well as moving it to the desired PoI in the city without energy overheads.

• **Research Contributions.** The following studies and mathematical formulations are the key contributions of this paper:

- We perform experimental studies on multihop video-streaming on UAVs, investigating the impact of the aerial relays on the video quality and on the service lifetime. Based on our observations, we determine the maximum number of hops that are feasible for such device-to-device communications, and the average energy consumed by the UAV components as a sum total of its motion, video-recording, wireless communication cost.
- Using the experimentally observed energy costs, the bus routes and time-tables for a target city scenario, we demonstrate the feasibility of the charging opportunities at random PoIs spread across the city, and at different times of the day.
- We derive a mathematical solution for UAV swapping and charging operations, taking into account the temporal/spatial availability of the charging stations installed on buses, so that the PoI is reliably monitored. Specifically, we show that the UAV scheduler can be modeled as an instance of multi-commodity flow problems, and mathematically solved through Mixed Integer Linear Programming (MILP) techniques. We also propose a heuristic and compute its computational complexity.
- We validate the proposed solution via a comprehensive emulation using real data from experimental studies, and live transportation updates and routes from the city

of Bologna, Italy, providing insights on the impact of different parameters.

The rest of the paper is structured as follows. In Section II, we review the related work on multi-hop video-surveillance with service persistence requirements. In Section III we provide experimental results motivating the proposed architecture. In Section IV, we introduce the system model, the MILP-formulation and the heuristic. Performance results are reported in Section V. Conclusions follow in Section VI.

II. RELATED WORKS

Most existing deployments assume direct communication between the UAVs and the ground sinks, with limited experimental studies conducted so far on real-time aerial video streaming. The pioneering study in [16] studied path loss and the small-scale fading in air-to-ground links. [6] demonstrated empirically that the TDMA MAC protocol can better support video-streaming applications than a CSMA MAC when UAV relays are employed. An application-layer video-encoding rate is considered in [7]. A video-streaming experiment where the UAVs move according to a fixed circular pattern is described in [17]. We differentiate our approach from these works based on the need to guarantee continuous surveillance service with energy constraints. Service persistence cannot be achieved via battery-dependent video-encoding schemes or energy-efficient communication protocols (such as the ones surveyed in [5]), since their benefits are offset by the limited flying time. For this reason, we believe incorporating ground-based infrastructure able to schedule the UAVs for battery replacement or charging operations is needed [9] [11] [12] [13] [14].

The study in [10] describes an automated battery management system that is able to sustain a three hours mission, using three UAVs, each with ten minutes of flying time. In [18], the authors propose an off-line path planning algorithm that ensures each UAV is able to reach the ground BS and replace its battery. In [9] [12] and [11], the authors address the problem of providing continuous aerial coverage with energy scheduling through a Mixed Linear Integer Programming (MILP) model. The flexibility of mobile charging units is considered by [13] and [14]. [13] proposes a scheduling algorithm which minimizes the travel time of the UAVs and determines the optimal flight path to the ground charging stations. In [14], the path planning problem is transformed into a Travel Salesman Problem (TSP) and solved via heuristics. Our work address persistent UAV service via mobile charging stations, like [13] and [14]. At the same time, it has differentiating novelties like considering city-run mobile charging units (e.g. buses) with fixed trajectories that are independent of the UAV positions. We also include the bus waiting time and the charger-induced UAV mobility to create a richer, practical model.

III. PRELIMINARY EXPERIMENTS AND MOTIVATION

This Section provides experimental results that study the impact of multi-hop UAV relaying (Section III-A) and realistic charging opportunities via a public bus transportation system (Section III-B).

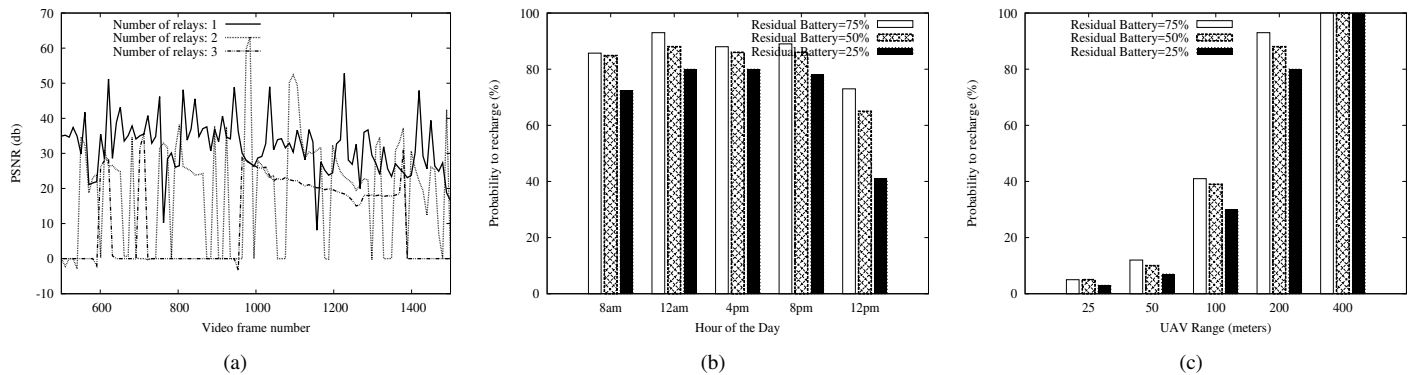


Fig. 2. The PSNR for aerial video-streaming applications, with different number of UAV relays, is depicted in Figure 2(a). The Recharge Probability at different hours of a work day and for different values of the visibility range are depicted in Figures 2(b) and 2(c) respectively.

Num Relays	Mean PLR (%)	Mean PSNR(db)
1 relay	18	32
2 relays	32	17
3 relays	38	6

TABLE I
PSNR AND PLR METRICS

A. Video-streaming over multi-hop UAV links

Our testbed is composed of four Parrot AR 2.0 GPS Edition UAVs equipped with a Raspberry PI 2.0 board, and a Wi-Fi radio module. In all our tests, one UAV captures the video feed and streams it toward a ground sink implemented on a laptop. We vary the number of equally spaced UAVs (8m) acting as static relays at the same uniform height between the source UAV and destination sink. We measure the performance of the multi-hop video communication, in terms of average Packet Loss Rate (PLR) and resulting Peak Signal to Noise Ratio (PSNR) using the well-known EvalVid tool-set for video analysis [20]. Figure 2(a) shows a trace segment of the PSNR values over time, while Table III-A reports the average PSNR values and packet loss rates computed over the full duration of the experiments. We see that one relay ensures satisfactory video-quality performance, while the PSNR for the configuration with two and three relays is 50% less. This shows that despite the extended reach, using more than one relay is unsuitable for practical surveillance applications.

B. Charging opportunities via public bus network

We consider the urban downtown area of Bologna, Italy, that is approximately 3 km², which is served by around 40 bus lines. The experiments leverage on the real locations of the bus stops, and the time-tables provided by the open-data service of the Bologna Passenger Transportation company (TPER). We model the scenario of a UAV recharging via an inductive wireless energy transfer station mounted on top of the buses (each bus is provided with one station). For 1000 random locations, we simulate the mobility of the UAV from those locations towards a bus stop within R meters (called as *visibility range*). If several choices are available, the UAV selects the stop with minimal energy discharge, considering

(i) the energy lost in order to reach the bus stop, and (ii) the energy cost of hovering over the stop before the bus arrives. The UAV parameters (speed, initial battery, discharge factor while flying) are modeled using the specifications of the Parrot AR 2. Figure 2(b) depicts the Recharge Probability, i.e. the percentage of times the UAV is able to recharge on the bus without running out of battery before its arrival, at different hours of the day (on the x-axis) and the initial residual energy (shown by bars). The range R is set to 200 meters. We find this probability is always higher than 70%, except for the night when the frequency of bus rides considerably decreases. Similarly, Figure 2(c) depicts the recharging probability when varying the R size, at a time setting of 12am. When considering $R=400$ meters, the charging opportunities are always guaranteed regardless of the initial position of the UAVs.

IV. BUS ROUTE OPTIMIZATION FOR UAVS

First, we define the system model and its main objectives for assigning UAVs to specific PoIs and bus routes (Section IV-A). Then, we address the problem via Mixed Integer Linear Programming (MILP) (Section IV-B). Finally, we provide a heuristic solution with complexity analysis (Section IV-C).

A. System Model

Video-streaming content is delivered by the UAVs to a central controller via fixed terrestrial WiFi Access Points (APs), scattered in the city. We consider a time slotted model $T=t_0, t_1, \dots$, with constant slot length equal to t_{slot} . The scenario can be defined as the tuple $\langle U, A, P, F, BR \rangle$, where:

- $U=\{u_1, u_2, \dots, u_{|U|}\}$ represents the set of available UAVs. Each UAV is equipped with GPS and WiFi modules, and provided with an initial energy amount equal to E_{init} .
- $A=\{a_1, a_2, \dots, a_{|A|}\}$ represents the set of WiFi APs.
- $P=\{p_1, p_2, \dots, p_{|P|}\}$ is the set of POIs that must be covered by the UAVs. We consider three categories of PoI: (i) PoI where video-surveillance is needed (denoted as Cat_1); (ii) PoI where a video-relay is needed (denoted as Cat_2), in order to connect a Cat_1 PoI to the closest AP; (iii) PoI where the UAV must act both as relay and video-streamer (denoted as Cat_3). Let $\nu(p_i)$ indicate the

category of PoI p_i . Based on the experimental results provided in Section III-A, we limit the network topology to a maximum of 2-hops (i.e., 1 intermediate UAV relay) between PoI and APs, in order to guarantee a satisfactory video quality (see Section III-A).

- $F = \{f_1, f_2, \dots, f_{|F|}\}$ is the set of bus-stops that can be served by multiple bus routes.
- $BR = \{b_1, b_2, \dots, b_{|BR|}\}$ is the set of bus routes defined by the stops and arrival/departure times. Thus, each route b_i is a couple $\{(f_S, t_E), (f_D, t_A)\}$, where $f_S, f_D \in F$ represent the source/destination stops, respectively, and $t_E, t_A \in T$ are the time slots of departure/arrival. Each route b_i is also associated with a cost $c(b_i) = t_A - t_E$, which represents the time required by the bus to cover the end-to-end path from f_S to f_A .

We denote $BR(f_j, t_k) \subseteq BR$ as the subset of routes that are available at time t_k from the source stop f_j , i.e. $BR(f_j, t_k) = \{b_i \in BR | b_i = \{(f_j, t_k), (f_*, t_*)\}\}$. For each PoI p_i , let $FS(p_i)$ be the *feasible stops* set of the f_j reachable from p_i within a time-slot, i.e. $FS(p_i) = \{f_j \in F | d(p_i, f_j) \leq R\}$, where $d(\cdot, \cdot)$ computes the Euclidean distance between two 3D points, and R is the visibility range given in Section III-B. Let $P(f_j) \subseteq P$ be the set of POIs reachable in a time-slot from f_j . Based on the previous definition we assume that an UAV located in any $f_j \in FS(p_i)$ can cover p_i . At each time slot t_k , each UAV u_i is in one of the following two states:

- *flying*, i.e. staying over a bus stop f_j and draining a constant per-slot amount of energy equal to $\beta \cdot t_{slot}$. We introduce the following binary *position variable* $x_{u_i, f_j}^{t_k} \in \{0, 1\} \forall u_i \in U, f_j \in F, t_k \in T$ such that:

$$x_{u_i, f_j}^{t_k} = \begin{cases} 1 & \text{if UAV } u_i \text{ flies over } f_j \text{ in time } t_k \\ 0 & \text{otherwise,} \end{cases}$$

- *riding*, i.e. recharging on top of a bus stop, and gaining a constant per-slot amount of energy equal to $\gamma \cdot t_{slot}$. We introduce the following binary *movement variables* $v_{u_i}^{t_k} \in \{0, 1\} \forall u_i \in U, t_k \in T$ such that:

$$v_{u_i}^{t_k} = \begin{cases} 1 & \text{if UAV } u_i \text{ "takes" a bus in time } t_k \\ 0 & \text{otherwise,} \end{cases}$$

Additionally, a UAV u_i in *flying* state over f_j can decide to cover a specific PoI $p_j \in P(f_j)$. In such case, the UAV will drain an additional per-slot amount of energy equal to $\alpha[\nu(p_j)] \cdot t_{slot}$, where $\alpha[\cdot]$ is the energy consumption depending on the coverage mode (i.e. Cat_1, Cat_2 or Cat_3), and whose values are set according to the experimental results of Section III-A. We introduce the following binary *coverage variable* $y_{u_i, p_j}^{t_k} \in \{0, 1\} \forall u_i \in U, p_j \in P, t_k \in T$ such that:

$$y_{u_i, p_j}^{t_k} = \begin{cases} 1 & \text{if UAV } u_i \text{ covers PoI } p_j \text{ in time } t_k \\ 0 & \text{otherwise,} \end{cases}$$

Let $RE(u_i, t_k)$ be the residual energy of UAV u_i at time slot t_k . Based on the state variables defined above, $RE(u_i, t_k)$

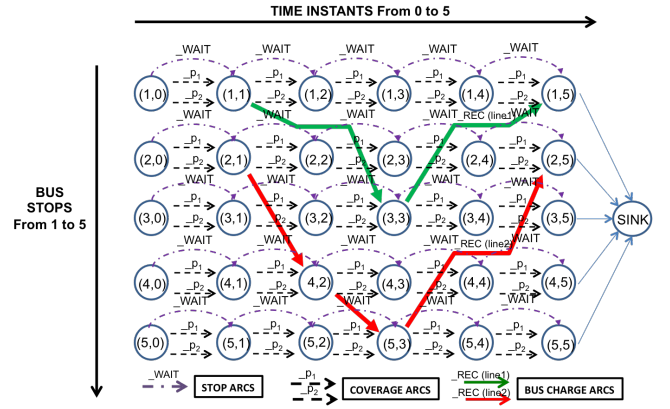


Fig. 3. Multiperiod directed multigraph for $|F|=5$ and $|P|=2$.

is updated at each $t_k \in T, \forall u_i \in U, \forall p_j \in P$, as follows:

$$\begin{aligned} RE(u_i, t_k) &= RE(u_i, t_{k-1}) \\ &- y_{u_i, p_j}^{t_k} \cdot \alpha[\nu(p_j)] \cdot t_{slot} \\ &- x_{u_i, f_j}^{t_k} \cdot \beta \cdot t_{slot} \\ &+ v_{u_i}^{t_k} \cdot \gamma \cdot t_{slot} \end{aligned} \quad (1)$$

Based on the $RE(u_i, t_k)$ values, we introduce the following additional decision variables:

- binary *UAV status-off variables* $z_{u_i}^{t_k} \in \{0, 1\} \forall u_i \in U, t_k \in T$ such that:

$$z_{u_i}^{t_k} = \begin{cases} 1 & \text{if UAV } u_i \text{ is off in } t_k, \text{ i.e. } RE(u_i, t_k) \leq 0 \\ 0 & \text{otherwise,} \end{cases}$$

- binary *global status-off variables* $w^{t_k} \in \{0, 1\} \forall t \in T$ such that:

$$w^{t_k} = \begin{cases} 1 & \text{if any UAV is off in } t_k \\ 0 & \text{otherwise,} \end{cases}$$

The system lifetime L is defined as the maximal time slot such that $L < t_k \cdot w^{t_k}$. Informally, the problem can be defined as: determine the variable states $x_{u_i, f_j}^{t_k}, y_{u_i, p_j}^{t_k}, v_{u_i}^{t_k}$ for each UAV u_i and time slot t_k , so that L is maximized, while each PoI p_j is continuously covered. In the following, we prove that the problem is a special instance of multi-commodity flow model, which is solved via MILP techniques.

B. MILP-based formulation

We devise a multiperiod directed multigraph $G(V, A)$ in order to model the movement of UAVs across the target area containing PoIs and bus stops. The set of nodes V is composed by the couple $(f_j, t_k), \forall f_j \in F$ and $\forall t_k \in T$. We denote a generic travel path as a so called *arc* e_i represented by $[(f_j, t_k), (f'_j, t'_k)]_l$, where (f_j, t_k) and (f'_j, t'_k) are, respectively, the tail and the head of the arc. Each arc also has an associated weight $W(e_i)$ and a subscript l that denotes one of three possibilities:

- Case 1. A single *stop arc* $[(f_j, t_k), (f_j, t_{k+1})]_{\text{WAIT}}$, which models the behavior of an UAV which is flying over stop f_j at time t_k , and will remain on the same stop also at

t_{k+1} (but without covering any POI). Here, the weight is given by the energy lost for flying, i.e., $W(e_i) = -\beta \cdot t_{slot}$;

- Case 2. Any one selection from the set of *coverage arcs* $[(f_j, t_k), (f_j, t_{k+1})]_{p_i}$, one for each $p_i \in P(f_j)$, modeling the behavior of an UAV covering the PoI p_i from stop f_j at time t_k , and will perform coverage of the same POI also at t_{k+1} . Here, the weight is given by the energy lost for surveillance based on PoI type, i.e., $W(e_i) = -(\alpha[\nu(p_j)] + \beta) \cdot t_{slot}$;
- Case 3. Any one selection from a set of *bus-charge arcs* $[(f_j, t_k), (\bar{f}, \bar{t})]_{REC}$, one for each $b_i = \{\bar{f}, \bar{t}\} \in BR(f_j, t_k)$, modeling the UAV taking a bus ride from the stop f_j at t_k , and charging on the top of the bus, until the next stop \bar{f} with arrival time \bar{t} . Here, the weight is given by the energy recharged during the ride, i.e., $W(e_i) = c(b_i) \cdot \gamma$, where $c(\cdot)$ is the cost function defined in Section IV-A.

For modeling ease, we add the following additional elements in the graph: (i) a sink node s , representing the destination of all flows, and (ii) a set of terminal arcs $[(f_j, L), s]$, $\forall f_j \in F$ connecting each terminal node to the sink s . The energy associated with such arcs is null, i.e. $W([(f_j, L), s]) = 0$. Figure 3 shows an example of a multiperiod directed multigraph built according to the rules defined above.

1) *Persistent Video Coverage Problem (PVCP)*: We model the action selection process of each UAV u_i through the following *non-splittable flow variables*:

$$\varphi_{[(f_j, t_k), (f_r, t_s)]_l}^{u_i} = \begin{cases} 1, & \text{if } u_i \text{ uses arc } [(f_j, t_k), (f_r, t_s)]_l \\ 0, & \text{otherwise} \end{cases}$$

Here, we want to determine the flow variables $[(f_j, t_k), (f_r, t_s)]_l$ for all UAVs to maximize the lifetime $L \in T$ so that $L < t_k \cdot w^{t_k}$. Additionally, the following constraints must be met:

$$\sum_{u_i \in U} y_{u_i, p_j}^{t_k} = 1 \quad \forall p_j \in P, t_k \in T \quad (2)$$

$$\sum_{f_j \in F} x_{u_i, f_j}^{t_k} + v_{u_i}^{t_k} \leq 1 \quad \forall u_i \in U, t_k \in T \quad (3)$$

$$y_{u_i, p_j}^{t_k} \leq \sum_{f_l \in F(p_j)} x_{u_i, f_l}^{t_k} \quad \forall u_i \in U, p_j \in P, t_k \in T \quad (4)$$

$$\sum_{[(\phi, \tau), (f_j, t_k)]_l \in A} \varphi_{[(\phi, \tau), (f_j, t_k)]_l}^{u_i} - \sum_{[(f_j, t_k), (\phi, \tau)]_l \in A} \varphi_{[(f_j, t_k), (\phi, \tau)]_l}^{u_i} = BAL_{(f_j, t_k)u_i}, \forall (f_j, t_k) \in V, u_i \in U \quad (5)$$

$$\sum_{[(f_j, t_k), (f_r, t_s)]_l \in A} \varphi_{[(f_j, t_k), (f_r, t_s)]_l}^{u_i} = x_{u_i, f_j}^{t_k} \quad \forall u_i \in U, l \neq REC, f_r \in F, t_s \in T \quad (6)$$

$$\sum_{[(f_j, t_k), (\bar{f}, \bar{t})]_{REC} \in RB(f_j, t_k)} \varphi_{[(f_j, t_k), (\bar{f}, \bar{t})]_{REC}}^{u_i} = v_{u_i}^{t_k} \quad \forall u_i \in U, f_j \in F, t_k \in T \quad (7)$$

$$RE(u_i, t_k) = RE(u_i, t_{k-1}) + \sum_{[(f_s, t_s), (f_j, t_k)]_l \in A} W([(f_s, t_s), (f_j, t_k)]_l) \cdot \varphi_{[(f_s, t_s), (f_j, t_k)]_l}^{u_i} \quad \forall u_i \in U, t_k \in T \quad (8)$$

$$z_{u_i}^{t_k} \leq z_{u_i}^{t_{k+1}} \quad u_i \in U, t_k \in T \quad (9)$$

$$w^{t_k} \leq w^{t_{k+1}} \quad t_k \in T \quad (10)$$

$$z_{u_i}^{t_k} \leq w^{t_k} \quad u_i \in U, t_k \in T \quad (11)$$

$$RE(u_i, t_k) + M \cdot z_{u_i}^{t_k} \geq 0 \quad \forall u_i \in U, t_k \in T \quad (12)$$

The constraints are explained as follows: (2) ensures that each PoI p_i is covered by exactly one UAV at any time $t_k \in T$; (3) says that at any $t_k \in T$, each UAV u_i must either be in *flying* state or in *riding* state; (4) ensures that a PoI $p_j \in P$ can be covered only by the UAV u_i located at a nearby bus stop $f_l \in F(p_j)$. Constraint (5) captured the (fundamental) flow conservation constraint. Here, the value of $BAL_{(f_j, t_k)u_i}$ depends on the considered node, i.e.:

- 1) for nodes $(f_j, t_0) \in V$, $BAL_{(f_j, t_0)u_i}$ is equal to -1 if the UAV u is in f_j at time t_0 (such nodes represent the sources of flows going through the network);
- 2) all nodes $(f_j, t_k) \in V$ such that $t_k \geq t_0$, have a null balance (i.e., $BAL_{(f_j, t_k)u_i} = 0$), since they must be just crossed by the UAVs;
- 3) the sink node s has the task of receiving all the flows sent through the network and thus $BAL_{s, u_i} = 1, \forall u_i$.

(6) imposes the following constraint: if a UAV u_i is flying over stop f_r in t_s (i.e., $x_{u_i, f_r}^{t_s} = 1$), then exactly one among the flow variables of u_i leading to vertex (f_r, t_s) must be active, i.e. equal to 1. (7) imposes a similar condition: if a UAV u_i takes a bus from f_j at t_k (i.e., $v_{u_i}^{t_k} = 1$), then exactly one among the flow variables of u_i associated with bus-charge arcs must be active. (8) is the energy update function, in accordance with (1). (9) states that any UAV u_i , which is unavailable at some t_k because it has drained all its residual energy, is also unavailable at t_{k+1} . (10) imposes the same condition on the system lifetime. (11) links the UAV and global status-off variable w_{t_k} . Thus, variable w_{t_k} is activated only when any UAV runs out of battery at time t_k . Finally, (12) ensures that the residual energy of each UAV is always greater than zero at each t_k , till the global status-off variable is activated (i.e. the system lifetime is reached); here, M is an arbitrary integer coefficient, much greater than E_{init} .

C. Heuristic solution of the MILP

The PVCP problem represents a complicated variant of an unsplittable multi-period multicommodity network design problem with side constraints, which is known to be NP-Hard (see e.g. [19]). In order to further reduce the computational complexity, we propose an heuristic approach that determines a polynomial time (though suboptimal) PVCP solution, and we analyze its performance later in Section V. The heuristic

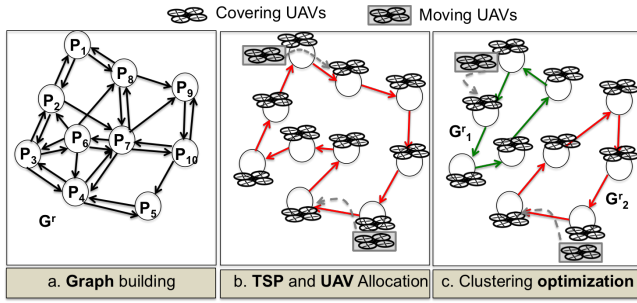


Fig. 4. Example of execution of the heuristic, considering a scenario with $|P|=10$ and $|U|=12$.

implements the UAV replacement mechanism such that at each time slot, each PoI is covered by exactly one UAV. The other UAVs (in number equal to $|U| - |P|$) move among the PoIs via the bus network, following a circuit which visits each PoI exactly once. Each time a UAV riding the bus reaches a PoI, it swaps places with the incumbent UAV, and the latter now moves to the next PoI of the circuit, using the bus-ride to recharge. We define the Replacement Interval (RI) index as the average amount of time which elapses between two consecutive replacements at a given PoI; clearly, the smaller RI index is, the lower is the energy consumed by UAVs for flying/coverage issues, and hence the higher is the system lifetime. Maximizing the RI index is hence translated into the dual problem of: (i) how to determine the optimal circuit; and (ii) where to initially place the non-covering UAVs within the circuit. For this purpose we propose a three phases algorithm: **Stage 1. Graph transformation:** First, we transform the multi-period directed multigraph $G(V, A)$ of Section IV-B into a new directed reduced graph $G^r(V^r, A^r)$, where the set of nodes V^r is the set of PoI P , and A^r is the set of arcs $a = [v_i^r, v_j^r]$ connecting two PoIs. Each arc has a cost $c^r([v_i^r, v_j^r])$ which represents the average minimum time required to travel from v_i^r to v_j^r using the bus transportation system. Figure 4.(a) shows the output of this phase (Section IV-C1 clarifies the operations enabling the graph transformation).

Stage 2. UAV path planning: Next, we compute the minimum cost circuit visiting each PoI exactly once and returning to the origin PoI using the Traveling Salesman Problem (TSP). Also TSP being NP-hard, several greedy heuristics are well known. We also allocate the non-covering UAVs at equal cost distance within the circuit, so that the RI index is minimized (see Figure 4.(b) and Section IV-C2).

Stage 3. Graph partitioning and optimization: Finally, we perform an optional optimization step, i.e. instead of considering a single circuit, we verify whether the RI index can be further improved by partitioning the graph $G^r(V^r, A^r)$ into several sub-graphs $G_1^r, G_2^r, \dots, G_q^r$. As before, we compute the TSP circuit and allocate the non-covering UAVs on each sub-graph (see Figure 4.(c)). The graph partitioning procedure stops when the RI index stops decreasing, and in any case has a fixed number of iterations, as explained in Section IV-C3. Algorithm IV-C shows the pseudo-code of the proposed

heuristic, including all the three stages which are further explained below.

Algorithm 1 PVCP Heuristic algorithm

```

1: Step 1. Build graph  $G^r(V^r, A^r)$  from  $G(V, A)$ 
2: for all arcs  $a^r = [v_i^r, v_j^r] \in A^r$  do
3:   for all time slots  $t_k$  in  $t_s \dots t_{s+h}$  do
4:     Set  $minTime=0$ 
5:     for all stops  $f_j \in FS(v_i^r)$  do
6:       Compute BFS over graph  $G$  from node  $(f_j, t_k)$ 
7:       if (meet node  $(f_l, t_m), f_l \in FS(v_j^r)$ ) and  $(t_m - t_k <$ 
8:          $minVal)$  then
9:         Set  $minTime=t_m - t_k$ 
10:        end if
11:       end for
12:       Add  $minVal$  to  $MTDS(v_i^r, v_j^r)$ 
13:     end for
14:   Set  $c^r([v_i^r, v_j^r]) = avg(MTDS(v_i^r, v_j^r))$ 
15: end for
16: Step 2. TSP and UAV Allocation
17: Compute circuit  $L(G^r)$  via TSP greedy heuristic
18: Place one UAV at each PoI  $p \in P$ 
19: Place the other  $|U| - |P|$  UAVs on PoIs at equal cost distance
20: Set  $worst(0) = \frac{c(L(G^r))}{|U| - |P|}$ 
21: Set  $solution=L(G^r)$ 
22: Step 3. Clustering optimization
23: for  $q = 1$  to  $|U| - |P|$  do
24:   Partition  $G^r$  into  $G_0^r \dots G_q^r$  through Kruskal clustering
25:   Allocate one UAV to each PoI  $p \in P$ 
26:   Allocate one UAV to each  $G_j^r \forall 0 \leq j < q$ 
27:   Set  $M(G_j^r) = 1 \forall 0 \leq j < q$ 
28:   for all the remaining  $|U| - |P| - q$  UAVs  $u_i$  do
29:     Compute  $RI(L(G_j^r)) = \frac{c(L(G_j^r))}{M(G_j^r)} \forall 0 \leq j < q$ 
30:     Extract  $G_j$ , with  $j = argmax RI(L(G_j^r)) \forall 0 \leq j < q$ 
31:     Add  $u_i$  to  $G_j$ , and place the UAVs at equal cost distance
32:     Update  $M(G_j^r) = M(G_j^r) + 1$ 
33:   end for
34:   Compute  $worst(q) = max(RI(L(G_j^r)) \forall 0 \leq j < q$ 
35:   if  $worst(q) < worst(q-1)$  then
36:     Set  $solution=\{G_0^r \dots G_q^r\}$ 
37:     Update  $q=q+1$ 
38:   else
39:     return solution
40:   end if
41: end for

```

1) *Stage 1. Graph transformation:* Differently from graph $G(V, A)$, the reduced graph $G^r(V^r, A^r)$ does not include the notion of bus time-table, but only about the path between PoIs and their average costs. For this reason, we assume that the reduced graph G^r is computed at different time segments of the day: each segment starts at t_s and ends at t_{s+h} , and the frequency of buses at each stop is assumed fixed or with minimum variations over each time segment. We approximate the cost $c^r([v_i^r, v_j^r])$ as the average minimal time required to travel from PoI v_i^r to v_j^r , at the target time segment. The computation of such costs can be derived from the original graph G with some elaborations. More specifically, we consider a specific time slot t_k , with $s \leq k \leq s+h$. We prune all the nodes $(\cdot, t_l) \in V$ with $l > s+h$, the corresponding arcs, and all the non bus-charge arcs. Then, a Breadth-First Search (BFS) over graph G is performed from

source node (f_j, t_k) , and the minimum time required to travel from a feasible stop of PoI v_i^r to a feasible stop of PoI v_j^r is determined (lines 6 – 10). The $MTDS(v_i^r, v_j^r)$ set keeps track of all the minimum travel times from v_i^r to v_j^r , when varying the starting time t_k (line 11). Finally, the average value of $MTDS(v_i^r, v_j^r)$ is returned as $c^r([v_i^r, v_j^r])$ (line 13). In order to determine the complexity of the procedure, we consider a worst case scenario where $FS(v_i^r) \sim F, \forall v_i^r \in V^r$. In such case, we can notice from the pseudo-code that the complexity of the full procedure is: $O(|A_r| \cdot |F| \cdot h \cdot |V|)$.

2) *Stage 2. UAV path planning*: This step computes an approximated minimal circuit over the previously defined reduced graph G^r . To this purpose, we employ a simple greedy algorithm, which selects at each step the edge with minimal cost, excluding those that lead to already visited PoI. The complexity of such a procedure is $O(|P|)$. Clearly, any higher complexity TSP approximation procedure can be used for this purpose, clearly increasing the solution optimality, but without affecting the behavior of the PVCP algorithm. Let $L(G^r)$ be the circuit over the PoI computed by the greedy algorithm, and $c(L(G^r))$ its total cost in terms of travel time. Moreover, let $RI(L(G^r))$ be the Replacement Interval index computed over the circuit $L(G^r)$. The next operation consists in defining the initial location of the UAVs, so that the RI index is minimized. First, one UAV is allocated to each PoI. If $|U|=|P|+1$, then the additional UAV is placed randomly among the PoI, and hence $RI(L(G^r)) = c(L(G^r))$. In the general case with $(|U| > |P| + 1)$, the $|U| - |P|$ additional UAVs are placed at equal cost distance, and hence $RI(L(G^r)) = \frac{c(L(G^r))}{|U|-|P|}$.

3) *Stage 3. Graph partitioning and optimization*: When the number of UAVs greatly exceeds the number of PoIs, the utilization of a single circuit may not be optimal in terms of the RI metric, specially in those use-cases where the bus network covers separate districts, each district is served by specific buses, and the distances between the different districts are significant. For this reason, a final optimization phase iteratively splits the graph into multiple clusters and then repeats Step 2 on each cluster. At each iteration q , the graph G^r is partitioned into q disjoint sub-graphs $G_1^r, G_2^r \dots G_k^r$ by using the Kruskal clustering algorithm, such that $\bigcup_k G_k^r = G^r$. The Kruskal algorithm ensures that the inter-cluster distance is maximized, hence it places nearby PoI reachable via bus on the same cluster. Then, the greedy TSP solution is computed over each G_i^r , and the initial placement of the UAVs is decided so that: (i) each PoI is covered by exactly one UAV; (ii) each sub-graph G_i^r contains at least one UAV; (iii) the minimum RI index of the worst sub-graph is maximized (lines 27-32). To this aim, we introduce the $worst(q)$ variable as proxy of the maximum RI index at iteration q (line 33). If the RI index decreases compared to the previous iteration $q-1$ (i.e. $worst(q) < worst(q-1)$ at line 34), then a further partitioning into $q+1$ sub-graphs is performed. Otherwise, the solution produced at iteration $q-1$ is returned. Note that the maximum number of iterations is bounded and cannot exceed $|U| - |P|$, since at least one UAV must be allocated to each sub-graph. In order to compute the complexity of

this stage, at each iteration q , we note that the TSP greedy algorithm is executed exactly q times, with $1 \leq q \leq |U| - |P|$. Hence, the worst-case complexity of this procedure is equal to: $\frac{(|U|-|P|)(|U|-|P|+1)}{2} \cdot |P| \sim O(|U|^2 \cdot |P|)$.

V. SIMULATION RESULTS

We evaluate the performance of our proposed PVCP heuristic through a practical city-scale simulation framework implemented in OMNeT++. All the characteristics of the scenario described in Section III-B are incorporated, including the video-surveillance application and the UAV operational functions, modeling the four states of: charging, discharging, waiting, and riding. The simulator accesses live transportation data provided by the city using published web-API. We implemented and compared the following algorithms:

- A basic *no-charging policy* -called NoRec (NR) in the following- where no charging infrastructure is available within the scenario. This scheme represents the lower bound on the performance of the bus-enabled charging policies, and is also the state-of-the art solution for most of current UAV deployments.
- An energy-centered *greedy policy* -called WORSTUAV (WU)- where the swap operations at a given PoI prioritize the UAVs with lowest energy levels. Here, the UAV not used for coverage selects as its destination the PoI served by the UAV with lowest residual energy, and then rides the bus network towards it.
- A distance-centered *greedy policy* -called BESTPOI (BP)- that prioritizes the distances to the PoI. After each swap operation at a given PoI, the UAV presently not participating in video coverage selects a subsequent PoI destination that is closest to its present location.
- A *TSP-based policy* -called TSP- that implements the full heuristic from Section IV-C, including the graph partitioning step.

We rigorously analyze the above policies using the following performance metrics:

- *System lifetime* (SL), defined from Section IV-A as the time instant when the first UAV runs out of battery capturing the persistence of the video-surveillance application.
- *Replacement counter* (RC), defined as the total number of UAV swap operations during the simulation. This metric measures how well the UAVs utilize the bus routes in the city, but also reflects the number of handovers that occur at application layer during the filming operations.
- *UAV state distribution* (USD), defined as the mean percentage of time that a UAV spends on covering PoI, waiting at the stop and riding the bus.

Unless specified otherwise, we used the following setting of the system parameters related to the UAV energy charge/discharge: $E_{init}=130000\text{J}$, $\beta=-100\text{W}$, $\alpha[Cat_1]=-120\text{W}$, $\alpha[Cat_2]=-150\text{W}$, $\alpha[Cat_3]=-170\text{W}$; $\gamma=25\text{W}$.

Figures 5(a), 5(b) and 5(c) show respectively the SL, RC and USD metrics when considering a number of PoI equal to 7 (i.e. $|P|=7$), and varying the number of UAVs, with $|U| > |P|$.

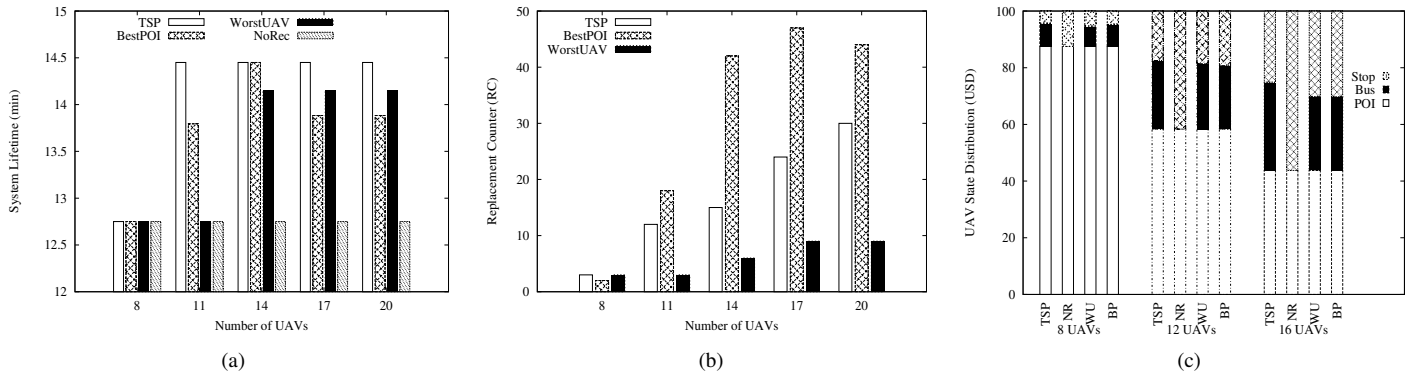


Fig. 5. The SL, RC and USD metrics for varying number of UAVs and fixed number of PoI are depicted in Figures 5(a), 5(b) and 5(c), respectively.

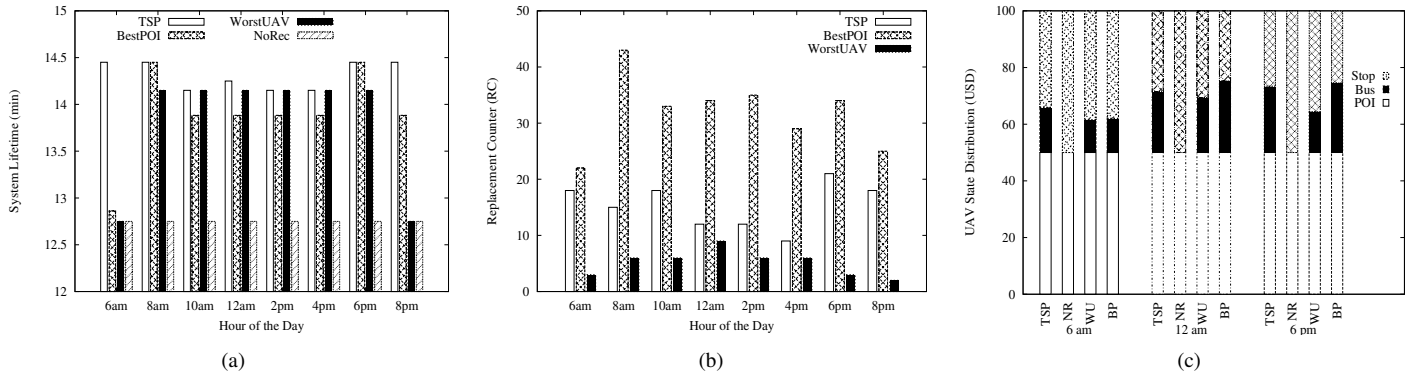


Fig. 6. The SL, RC and USD metrics for fixed number of UAVs and number of PoI, at different hours of the day, are depicted in Figures 5(a), 5(b) and 5(c), respectively.

We generate random locations of the PoI within the scenario for each test, and then average the results. Figure 5(a) shows that our solution outperforms the NR, WU and BP algorithms in terms of SL metric for $|U| > 8$, with an average gain of around +18% compared to the no-recharge case. Vice versa, the WU algorithm produces almost the same results as the basic NR policy, since the PoI selection prioritizes the energy factor, but without considering the distance and the bus tables; hence, a UAV willing to move toward a PoI where the current UAV is running out of battery, might also completely drain its battery while waiting at a bus stop. Figure 5(b) shows the RC metric for the same scenario. We can notice that the BP policy incurs an average number of UAV swaps much greater than our solution. Each swap is an overhead from an end-user application perspective, since interruptions during a handoff degrade the video quality. Vice versa, the TSP solution limits the number of UAV replacements, while still guaranteeing a longer lifetime than the BP policy. Finally, Figure 5(c) shows the USD metric, for three configurations of $|U|$. Here, we report the average fraction of time spent by the UAV in each of the three states: (i) covering a PoI (denoted with the POI label in the Figure), (ii) waiting at a bus stop (denoted with the Stop label) or (iii) moving/recharging by riding on top of a bus (denoted with the Bus label). When increasing the number of available UAVs (e.g. from 8 to 16), the average

time spent in the bus riding state increases significantly for the TSP algorithm, since less UAVs are required for PoI coverage, and the remaining ones travel using the ground transportation system. This further justifies the SL improvement of the TSP algorithm, when increasing the number of UAVs from 8 to 16. Figures 6(a), 6(b) and 6(c) show respectively the SL, RC and USD metrics for a fixed number of PoI ($|P|=7$) and UAVs ($|U|=14$), but considering different temporal availability of the buses at each stop. Again, Figure 6(a) shows that the TSP solution is able to maximize the SL metric compared to the other policies. It is slightly affected by the bus frequency, with an average gain of +15% compared to the no-recharge case. Figure 6(b) shows the RC metric over the same scenario. As before, the BP policy incurs in much higher number of UAV replacements, but without leading to an improvement in terms of lifetime and adversely affecting the quality of video-streaming. Figure 6(c) depicts the USD metric at three hours of the day. Comparing the cases of 6am and 6pm for the TSP policy, we notice that the UAVs increase the fraction of time in Bus riding state compared to the POI state: this further justifies the ability of the proposed solution to adapt the UAV deployment to the real-time bus schedules. Finally, we complete our study by providing further insights about the performance of individual stages of the proposed TSP heuristic. We consider an extended

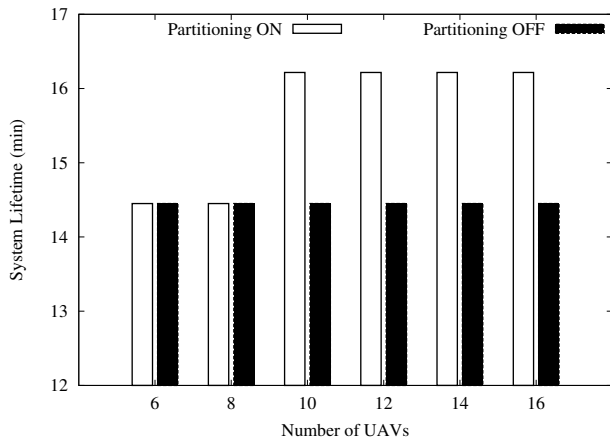


Fig. 7. The SL metric when enabling/disabling the graph partitioning stage of the TSP algorithm, for a two-district city scenario.

scenario of around 5 km^2 , which includes the downtown area (as is present in the simulations so far) as well as one of the major suburban districts. This district is connected to the downtown area via a limited number of inter-district bus routes, but also has an additional number of buses that serve exclusively the local area. Figure 7 compares the SL metric for two different versions of the TSP algorithm, i.e. (i) the No-partitioning case, where the stage 3 of the algorithm is not executed (hence, only a single TSP circuit is completed), or (ii) the Partitioning-enabled case, where the stage 3 of the algorithm is executed as described in Section IV-C3. We vary the number of UAVs $|U|$, again for a fixed number of random PoI (equal to 7), considering the bus schedules at 6pm. We notice that the graph partitioning stage becomes effective only when the number of PoI is higher than a threshold, equal to eight in this case (otherwise, the constraint of UAV allocation at line 25 of Algorithm 1 is not satisfied). However, for $|U| > 8$, the optimization provides an additional +10% gain over the basic TSP algorithm.

VI. CONCLUSION AND FUTURE WORKS

In this paper, we designed an innovative system architecture for aerial video surveillance of city-wide PoI, which copes with the energy-limited flight times of the UAVs without requiring the installation of a fixed charging infrastructure. Through measurements and simulations with real city-provided transportation maps and schedules, we demonstrated the possibility for deploying mobile charging stations on top of urban buses while incurring 30-50% fewer energy-related UAV swap instances, when compared with schemes that prioritize distance to the closest PoI. Our simulation and analysis demonstrate that our proposed solution can be feasibly applied over large-scale city scenarios, and can achieve 15-20% lifetime extensions compared to the state-of-the-art solution, where UAVs are deployed without charging possibilities. Future works include: a stochastic problem formulation taking into account live traffic conditions and bus delays at each stop accompanied by implementation on a real testbed.

ACKNOWLEDGEMENTS

This work was supported in part by the U.S. Office of Naval Research under grant number N000141612651 and by the ALMAIDEA UniBO Grant Senior Project: “BeeDrones: environmental monitoring systems based on ultra-low power sensor nodes and aerial communications”. It has been carried out in the framework of the DIVINA Challenge Team, which is funded by the Labex MS2T program. Labex MS2T is supported by the French Government, through the program “Investments for the future”, managed by the French National Agency for Research (Reference ANR-11-IDEX-0004-02).

REFERENCES

- [1] Federal Communication Commission (FCC). Deployable Aerial Communications Architecture in Emergency Communications. 2011.
- [2] Gartner Inc. <http://www.gartner.com/newsroom/id/3602317> 2017.
- [3] M. Erdelj, E. Natalizio, K. R. Chowdhury and I. F. Akyildiz Help from the Sky: leveraging UAVs for disaster management. *IEEE Pervasive Computing*, 16(1): 24-32 (2017).
- [4] M. Erdelj, O. Saif, E. Natalizio, I. Fantoni UAVs that fly forever: Uninterrupted structural inspection through automatic UAV replacement *Ad Hoc Networks*, Available online 12 December 2017.
- [5] Y. Zeng, R. Zhang and T. Joon Lim. Wireless communications with unmanned aerial vehicles: opportunities and challenges. *IEEE Communication Magazine*, 54(5), pp. 36-42, 2016.
- [6] L. R. Pinto, L. Almeida and A. Rowe. Video streaming in multi-hop aerial networks. *Proc. of IEEE INSPN, Pittsburgh, USA, 2017*.
- [7] R. Muzaffar, V. Vukadinovic and A. Cavallaro. Rate-adaptive multicast video streaming from teams of micro aerial vehicles. *Proc. of IEEE ICRA, Stockholm, Sweden, 2016*.
- [8] J. Leonard, A. Savvaris and A. Tsourdos. Energy management in swarm of Unmanned Aerial Vehicles. *Proc of IEEE ICUAS, Atlanta, USA, 2013*.
- [9] J. Kim, B. D. Song and J. R. Morrison. On the scheduling of systems of UAVs and fuel service stations for long-term mission fulfillment. *Journal on Intelligent Robot Systems*: 70(1), pp. 347-359, 2013.
- [10] N. K. Ure, G. Chowdhary, T. Toksoz, J. P. How, M. A. Vavrina and J. Vian. An automated battery management system to enable persistent missions with multiple aerial vehicles. *IEEE/ASME Transactions on Mechatronics*, 20(1), pp. 275-286, 2015.
- [11] B. D. Song, J. Kim, H. Park, J. R. Morrison and D. H. Shim. Persistent UAV service: an improved scheduling formulation and prototypes of system components. *Journal of Intelligent & Robotic Systems*: 74(1), pp. 221-232, 2014.
- [12] N. Nigam, S. Bicinawski, I. Kroo and J. Vian. Control of multiple UAVs for persistent surveillance: algorithm and flight test results. *IEEE Transactions on Control System Technology*, 20(5), pp. 1236-1251, 2012.
- [13] K. Yu, A. K. Budhiraja and P. Tokekar. Algorithms for routing of Unmanned Aerial Vehicles with mobile recharging stations and for package delivery. *International Symposium on Aerial Robotics*, Philadelphia, USA, 2017.
- [14] N. Matheuw, S. L. Smith and S. L. Waslander. Multirobot rendezvous planning for recharging in persistent tasks. *IEEE Transactions on Robotics*, 31(1), pp. 128-142, 2015.
- [15] A. Shariat, A. Tizghadam and A. Leon-Garcia. An ICN-based publish-subscribe platform to deliver UAV service in smart cities. *Proc. of IEEE INFOCOM WORKSHOP, San Francisco, USA, 2016*.
- [16] E. Yanmaz, R. Kuschnig, and C. Bettstetter. Achieving air-ground communications in 802.11 networks with three-dimensional aerial mobility. *Proc. of IEEE INFOCOM, Turin, Italy, 2013*.
- [17] A. Abdel-Hadi, K. Michel, A. Gerstlauer and S. Vishwanath. Real-time optimization of video transmission in a network of AAVs. *Proc. of IEEE VTC Fall, San Francisco, USA, 2011*.
- [18] J. Scherer and B. Rinner. Persistent multi-UAV surveillance with energy and communication constraints. *Proc. of IEEE CASE, Forth Worth, USA, 2016*.
- [19] F. D’Andreagiovanni, J. Krolkowski, J. Pulaj. A fast hybrid primal heuristic for multiband robust capacitated network design with multiple time periods. *Applied Soft Computing* 26, pp. 497-507, 2015.
- [20] Jirka Klaue. EvalVid - A Video Quality Evaluation Tool-set Website: <http://www2.tkn.tu-berlin.de/research/evalvid/fw.html>