IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 7, 2019, accepted May 28, 2019, date of publication June 5, 2019, date of current version June 24, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2921269

Future UAV-Based ITS: A Comprehensive Scheduling Framework

HAKIM GHAZZAI[1], (Member, IEEE), HAMID MENOUAR[2], (Senior Member, IEEE), ABDULLAH KADRI[2], (Senior Member, IEEE), AND YEHIA MASSOUD[1], (Fellow, IEEE)
[1]Stevens Institute of Technology, Hoboken, NJ, USA
[2]Qatar Mobility Innovations Center (QMIC), Qatar University, Doha, Qatar

Corresponding author: Hakim Ghazzai (hghazzai@stevens.edu)

This work was made possible, in part, by grant NPRP#9-257-1-056 from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.

ABSTRACT Multi-rotor unmanned aerial vehicles (UAVs) have been recently recognized as one of the top emerging technologies to be utilized in various smart city domains such as intelligent transportation systems (ITS). They represent an innovative mean to complement existing technologies to surveil transportation network, control traffic, and monitor incidents. The UAVs usually operate for time-limited missions due to their limited battery capacity. Hence, they need to frequently return to their docking stations to recharge their batteries, which handicaps their mission coverage and performance. When designing a UAV-based ITS infrastructure, it is crucial to leverage the UAV fleet effectively. In this paper, a generic management framework of UAVs for ITS applications is developed. The problem of docking/charging station placement is first investigated to find optimized locations for a given number of stations to be installed by the ITS operator. To this end, two fundamental criteria are taken into account: i) the flying time required by the UAV to reach the mission/incident location and ii) the risk of battery failure during the UAV operation. The two algorithms, namely a penalized weighted k-means algorithm and the particle swarm optimization algorithm, are designed for this purpose. Once the docking stations are placed, a UAV scheduling program is formulated to optimally cover the pre-known missions while minimizing the total energy consumption of the fleet and respecting a coverage efficiency targeted by the ITS operator. The proposed proactive scheduling approach employs multiple UAVs in sequential and parallel ways to cover spatially and temporally distributed events in the geographical area of interest over a long period.

INDEX TERMS Intelligent transportation system, planning, scheduling, unmanned aerial vehicles.

I. INTRODUCTION

The usage of unmanned aerial vehicles (UAVs), *aka* drones, has drastically increased over the last decade, not only in military but also in public and civil domains. The constant increase in demand in the market for UAVs is due to their decreasing cost and the flexibility and dynamic capabilities they offer. Another trait includes the mounting of various devices such as cameras, sensors, and communication interfaces. UAVs is being utilized in a variety of different domains including agriculture, security and surveillance, delivery of goods and services, and telecommunications [2]. In the coming years, the multi-rotor UAVs market will witness unprecedented growth. According to the PwC report, the emerging global market for business using UAVs is estimated at over \$127 billion where infrastructure monitoring and transport are expected to share more than 45% of the total market. As an example, we can cite the Uber Air, which is developing a shared air transportation solution that is expected to be tested by 2020 and will present a stride in UAV technology [3].

The intelligent transportation systems (ITSs) community has also shown particular interest in the potential employment of such flying units to support existing technologies and address various ITS challenges [4]. The three-dimensional mobility of UAVs gives an additional degree of freedom, in comparison to the fixed road side units (RSUs), for different applications such as incident reporting and traffic monitoring. Indeed, the RSU, a ground device that functions as a communication mean to support transportation infrastructure and smart vehicles, can only detect incidents in areas within its specific range and is subject to placement restriction constraints. In contrast, UAVs have potentials to mitigate such challenges of area coverage size and road network restrictions and have the ability to fly at different altitudes which helps

The associate editor coordinating the review of this manuscript and approving it for publication was Dimitrios Katsaros.

in ameliorating their wireless channel qualities and increasing their communication ranges. UAVs would significantly enhance the signal strength and system throughput which enables multiple bandwidth-hungry applications such as the transmission of high-resolution images and videos [5]. Moreover, the adoption of UAVs would help operate ITS at a lower cost compared to RSUs, known to have high deployment and installation costs. Finally, UAVs could adequately function in real-time based on needs which allow for lower energy use and maintenance cost.

Nonetheless, UAVs face some challenges in the context of ITS which substantially limits its active operations [6]. For example, its limited battery capacity hinders its constant use. Admittedly, they have to continuously retreat to their docking station to recharge their batteries and be able to pursue their trips or missions. Hence, recurring to-and-fro trips transpire continuously, and an optimization approach is needed to manage the UAV energy power more efficiently. The UAV energy management needs to consider two main components. The first component is due to the primary hardware unit and its mobility, while the second component is related to any other component installed on the UAV such as sensors, camera, and/or the communication interfaces. Separately optimizing both energy components may lead to suboptimal results since they are correlated in many situations. For instance, reducing the traveled distance (i.e, flying energy) may require the positioning of the UAV at a location quite far from the target. This will lead to a longer transmission time to send packets which will consequently increase the communication energy. Therefore, a UAV scheduling management scheme must consider the different aspects related to the operation of the fleet.

Another challenge that might be confronted by UAVs is the mitigation of emergency conditions. Response time in such situations is crucial since the UAV must reach the incident location in the shortest time — for example, UAV is required to cover an accident and monitor evacuations by sending periodic photos or videos to operators to analyze the situation. Another example would be the monitoring of a jammed intersection and the quick deployment of flying traffic lights to decrease the delay of emergency vehicles, such as ambulances [4].

A. RELATED WORK

Most of the studies existing in the literature investigating the use of UAVs in ITS applications focus on either improving their operation efficiency in performing missions such as monitoring, tracking, and detection using UAV videos/imagery [7]–[9] or enhancing the data exchange through UAV-assisted vehicular ad hoc networks (VANets) [10], [11]. Other studies propose UAV deployment and trajectory optimization schemes but applied in other fields such as cellular networks [12]–[16]. However, these work completely or partially ignore the energy and time travel constraints in their models and focus rather on the UAV missions by itself. The aforementioned constraints are very essential to ensure the successful operation of UAVs in ITS and they closely depend on the following factors: i) the characteristics/specifications of the UAVs, e.g., maximum UAV speed, power consumption levels, battery capacity etc., ii) the followed trajectory and UAV stops if any, e.g., the trajectory and optimized 3D locations, and iii) the locations of the UAV docking stations where the UAV will need to go to refill its battery and where it will land during idle periods. Therefore, the optimization of the docking stations' locations in large geographical areas is the first step that should be pursued in designing efficient UAV systems for ITS applications. Then, other research activities can be conducted to improve the UAV operation such as positioning and path planning of UAVs. Once the docking station locations are determined, it becomes very important to optimize the UAV scheduling over a long period of time in order to select and determine which UAVs should be sent to execute a given task in a certain location of the area of interest. The UAV scheduling optimization is not a trivial task. It has to guarantee the smooth execution of the tasks without leading to redundant exploitation of the resources, e.g., energy and UAVs, especially for applications that may require the handoff of multiple UAVs. The scheduling decision requires the consideration of multiple input parameters influencing the whole system including the current positions of the UAVs, the amount of stored energy, the battery capacity, the spatial and temporal information about the missions to be executed, and the locations of the docking stations, etc. Most of the studies existing in the literature investigating the use of UAVs in ITS applications focus on either improving their operation efficiency in performing missions such as monitoring, tracking, and detection using UAV videos/imagery [7]–[9] or enhancing the data exchange through UAV-assisted vehicular ad hoc networks (VANets) [10], [11]. Other studies propose UAV deployment and trajectory optimization schemes but applied in other fields such as cellular networks [13]–[16]. However, these work completely or partially ignore the energy and time travel constraints in their models and focus rather on the UAV missions by itself. The aforementioned constraints are very essential to ensure the successful operation of UAVs in ITS and they closely depend on the following factors: i) the characteristics/specifications of the UAVs, e.g., maximum UAV speed, power consumption levels, battery capacity etc., ii) the followed trajectory and UAV stops if any, e.g., the trajectory and optimized 3D locations, and iii) the locations of the UAV docking stations where the UAV will need to go to refill its battery and where it will land during idle periods. Therefore, the optimization of the docking stations' locations in large geographical areas is the first step that should be pursued in designing efficient UAV systems for ITS applications. Then, other research activities can be conducted to improve the UAV operation such as positioning and path planning of UAVs. Once the docking station locations are determined, it becomes very important to optimize the UAV scheduling over a long period of time in order to select and determine which UAVs should be sent to execute a

given task in a certain location of the area of interest. The UAV scheduling optimization is not a trivial task. It has to guarantee the smooth execution of the tasks without leading to redundant exploitation of the resources, e.g., energy and UAVs, especially for applications that may require the hand-off of multiple UAVs. The scheduling decision requires the consideration of multiple input parameters influencing the whole system including the current positions of the UAVs, the amount of stored energy, the battery capacity, the spatial and temporal information about the missions to be executed, and the locations of the docking stations, etc.

The scheduling and management of UAV fleet have attracted a lot of research attention in the recent years. Several approaches based on different techniques such as traveling salesman problem [14], [17], bio-inspired algorithms [18], [19], and consensus-based grouping algorithm [20] have been developed. Such problems are classified as unmanned aerial vehicle routing problems (VRPs) and are generally designed to find optimized routes for UAVs to execute a list of missions. For instance, two multitrip VRPs dealing with the energy consumption issue are proposed in [21] for UAV-delivery applications. The first problem minimizes the total delivery cost under a certain delivery time limit, while the second problem focuses on the dual version to minimize the total delivery time with respect to a budget constraint. The problems are modeled as Mixed integer linear programs (MILP). Another VRP problem has been formulated using a graph-based MILP [22] to determine the minimum coverage time of ground areas using a fleet of fixed-wing UAVs equipped with imagers. The model considers the time needed to launch the UAVs but ignore the battery management of the flying vehicles.

Few work have focused on optimizing the UAV scheduling in its generic form or adapted for applications other than ITS. Most of the studies imposed certain assumptions to relax the problem formulation and simplify its resolvability. For example, in [23], the authors designed a centralized UAV management method for intelligent surveillance and reconnaissance missions. A MILP is formulated to determine the path followed by the UAVs to collect data from spatially distributed nodes and deliver them to a sink. The collection time is minimized given cycle length constraints. A dynamic programming approach for UAV delivery systems is provided in [24] where an optimization problem looking for the minimization of the service delivery delay is developed. Both of the earlier studies focus on the management of multiple UAVs. However, the adopted approaches do not consider temporal characteristics of the missions (starting time or duration) and/or do not take into account the limited battery capacity issue.

The studies presented in [25], [26] dealt with the limited energy challenge. The authors have formulated a MILP problem to model the UAV scheduling process. Each UAV is assumed to be associated to a specific mission characterized by a pre-known trajectory. Along the way, multiple charging stations are made available to reload the UAVs' batteries. When needed, UAVs can substitute each other to complete the missions. The problem is simplified by decomposing each mission into multiple sub-missions such that the instants of updating the statuses of the UAVs is based on the job split choice of the UAV operator. Tthe genetic algorithm is then employed to achieve a suboptimal solution of the problem. Similar formulation is developed in [27] but, heuristic approaches are employed to reduce the problem solving complexity while assuming random initial locations of the UAVs. Supposing pre-known trajectories and splitting them into multiple sub-missions simplify the problem formulation and leads to suboptimal solutions since the instants of decision making are neither time flexible nor optimized. In our previous work [28], we have designed a scheduling framework for UAVs where only one docking station is used to reload all the flying units.

B. CONTRIBUTIONS

The present paper develops a generalized framework optimizing the UAV operations for ITS applications. It first focuses on solving a planning problem to determine optimized permanent locations for docking stations to which UAVs need to go to recharge their batteries. Afterwards, the paper proposes a proactive UAV scheduling framework that optimizes the UAV operation and assign UAVs to multiple events identified by different spatial and temporal characteristics over long period of time.

The goal of the first phase is to determine where to place the docking stations in a given geographical area in order to maximize a coverage efficiency metric. The docking stations must be located near the areas that interest the ITS operator, e.g., traffic accident hot spots. The coverage efficiency metric is measured according to the historical accidents and traffic jam statistics of the road segments and intersections. The docking station planning problem is performed while considering i) the UAV travel time to reduce delayed deployment of the flying units during emergency situations and ii) the energy constraint to avoid battery depletion during the UAV operation. The proposed planning approach can be applied for UAVs mounting cameras or communication interfaces, e.g, for UAV-assisted VANets and is solved using the following two proposed algorithms: a penalized and weighted k-means clustering algorithm (PWkMeans) and an algorithm based on swarm intelligence, namely particle swarm optimization (PSO). The performances of both algorithms are provided by comparing the achievable coverage efficiency. Unlike similar studies investigating the deployment of RSUs [29], [30] where uniform grid maps are used, the proposed approach can be applied to any realistic map and does not necessary require pre-defined possible stations' locations (i.e., discrete optimization). The proposed approach can also consider the existence of already deployed RSUs

The second phase uses the outputs of the first phase to manage the operation of a fleet of UAV. It aims to assign them to multiple events/missions identified by their spatially distributed locations and time characteristics, i.e., starting instants and duration. The objective is to minimize the total energy consumption while ensuring the completion of all events and missions. In addition to the locations of the docking stations, the scheduling process will also need to consider the flying time required to move the UAV from one location to another as well as the charging time. Indeed, a UAV will not be available to execute a mission or cover an event unless it has sufficient energy to ensure at least the round trip from and to the docking station. Therefore, the proposed scheduling framework allows parallel execution of disjoint events and the sequential UAV operation while covering events and executing missions. The sequential coverage allows the substitution of one UAV by another when covering the same event. Moreover, the framework enables the transfer of UAVs from a docking station to another according to the future pre-planned missions. Hence, if an area surrounding docking station A is free of events, the UAVs will take off and fly to another docking station close to more important hot spots. A mixed integer nonlinear programming (MINLP) problem incorporating all these characteristics and properties is formulated and optimally solved. Four possible actions can be taken by the UAVs: covering an event, waiting at an event, flying, and charging battery. As outputs, the optimizer provides the starting times of each of these actions, their corresponding time duration for each UAV, and the locations of the UAVs at each instant. It is worth to note that UAVs can be used more than once during the time horizon and can successively serve multiple events according to the energy availability. In case of energy depletion, UAVs will return to one of the docking stations to recharge their batteries.

C. PAPER OUTLINE

The paper is organized as follows. Section II presents the system model. The problem of docking station placement is investigated in Section III. In Section IV, the MINLP problem for UAV scheduling and the steps to solve it are presented. Selected simulation results are provided in Section V. Section VI discusses the perspective and open challenges of the use of UAV in ITS. Finally, concluding remarks are drawn in Section VII.

II. SYSTEM MODEL

We consider to deploy a UAV-based traffic monitoring system for a road network in a given geographical area characterized by N points of interest (PoI) that can represent intersections, streets, or segments of road. Each PoI n, where $n = 1, \ldots, N$ is defined by the latitude and longitude coordinates, denoted by (ϕ_n, ψ_n). We suppose that a UAV will leave a docking station to fly towards a sub-area to cover a certain number of these N PoIs characterized by a certain coverage efficiency level of a UAV depends on its characteristics, its final location, and the locations of its docking station. We consider that the ITS operator aims to install S docking stations at locations (ϕ_s, ψ_s) to be determined where $s = 1, \ldots, S$. For simplicity and without loss of generality, we assume that the altitudes h_n of the N PoIs and h_S of the S docking stations are set to zero $h_n = h_u = 0, \forall s, n$. In the following, we introduce the power model as well as the coverage of a UAV d, with $d = 1, \ldots, D$ and D is the total number of UAVs used by the operator, located at a position u defined by the 3D geographical coordinates (ϕ_u, ψ_u, h_u).

A. UAV POWER MODEL

The power model of a UAV is composed by two components: hovering and hardware power and power related to the mounted devices. We assume that the operator is using UAVs having similar characteristics except the average speed, denoted by V_d. The hover and hardware power levels, denoted by P^{hov} and P^{har}, can be expressed, respectively, as [31][1]:

$$P^{\text{hov}} = \sqrt{\frac{(m_{\text{tot}} g)^3}{2\pi r_p^2 n_p \rho}}, \quad \text{and } P_d^{\text{har}} = \frac{P^{\text{full}} - P^s}{v^{\max}} V_d + P^s, \quad (1)$$

where m_{tot}, g, and ρ are the UAV mass in (Kg), earth gravity in (m/s^2), and air density in (Kg/m^3), respectively. The parameters r_p and n_p denote the radius and the number of the UAV's propellers, respectively. The maximum UAV speed is denoted by $v^{\max}$ while P^{full} and P^s are the hardware power levels when the UAV is moving at full speed and when the UAV is in static mode, respectively. Finally, the motion power of a UAV can be calculated as:

$$P_d^{\text{mot}} = P^{\text{hov}} + P_d^{\text{har}}. \quad (2)$$

Accordingly, the total flying energy required by a UAV to fly from a docking station s to a specific location u is expressed as follows:

$$E_d^{\text{mot}}(s, u) = P_d^{\text{mot}} T_d^{\text{fly}}(s, u),$$
$$\text{where } T_d^{\text{fly}}(s, u) = \frac{\sqrt{\text{Dist}(s, u)^2 + h_u^2}}{V_d}, \quad (3)$$

where $\text{Dist}(s, u)$ is the distance separating s and u and computed using the haversine formula as follows:

$$a = \sin^2\left(\frac{\Delta\phi_{s,u}}{2}\right) + \cos(\phi_s)\cos(\phi_u)\sin^2\left(\frac{\Delta\psi_{s,u}}{2}\right),$$
$$c = 2\,\text{atan2}(\sqrt{a}, \sqrt{1-a}),$$
$$d(s, u) = R\,c, [km] \quad (4)$$

where $\Delta F_{i,j} = F_i - F_j$ with $F \in \{\phi, \psi\}$ and $i, j \in \{s, u\}$, atan2 is the inverse tangent function with two arguments, and R is the earth Radius ($R = 6371$ km). In (3), the term $T_d^{\text{fly}}(s, u)$ indicates the time needed for UAV d to fly from location s to location u.

On the other hand, the UAV consumes extra energy related to the mounted device. For instance, the communication interface power can be linearly modeled as follows [33]:

$$P^{\text{com}} = \gamma P_u^{\text{tx}} + \delta, \quad (5)$$

[1]Other UAV power models existing in literature such as [32] can be used with this framework. Indeed, the developed framework can work with any power model and does not require a specific formulation as long as the average UAV speed is considered.

where γ is a scaling parameter for the radiated power P_u^{tx} and δ models a constant power due to signal processing hardware independent. The value of P_u^{tx} depends on the location u of the UAV as it will be shown in the sequel.

B. UAV COVERAGE

We consider that a UAV can successfully execute a mission if a PoI is within its maximum range. For example, if the UAV is charged to collect images or videos, its maximum range or coverage is directly related to the see-ability/visibility of the UAV's camera which depends on the quality of the camera and the UAV location [34]. In this paper, we are interested in UAVs equipped with communication interface. The coverage, in this case, is represented by a surface πr_u^2 where r_u is the radius of a circle centered at the UAV projection onto the 2D earth surface, i.e., (ϕ_u, ψ_u). Within this range, all smart vehicles and RSUs can successfully decode the UAV signal and vice versa. The detection efficiency is subject to the wireless channel quality between the flying and the ground terminals which is, on average, depending on the path loss (PL) effect. The PL of the air-to-ground link is a weighted combination of two PL links: LoS and non-LoS (NLoS) links. It is shown that obtaining a LoS is not always possible even for flying UAV Indeed, there is a probability to obtain a LoS link between the UAV and a ground terminal which is based on the UAV's altitude, the distance separating the two transceivers, and the environment [35]. The average PL, in dB, between the UAV positioned at a position (ϕ_u, ψ_u, h_u) and a ground terminal located at a position (ϕ_v, ψ_v) in urban environments is given as [35]:

$$\overline{PL}_{u,v} = p_{u,v}^{\text{LoS}} PL_{u,v}^{\text{LoS}} + (1 - p_{u,v}^{\text{LoS}}) PL_{u,v}^{\text{NLoS}}. \tag{6}$$

where $p_{u,v}^{\text{LoS}}$ is the LoS probability given by:

$$p_{u,v}^{\text{LoS}} = \frac{1}{1 + \alpha \exp(-\beta[\theta_{u,v} - \alpha])}, \tag{7}$$

where $\theta_{u,v}$ is the elevation angle between the UAV u and the ground terminal v in degree and is evaluated as $\theta_{u,v} = \text{atan}\left(\frac{h_u}{d(u,v)}\right)$. The parameters α and β are two constant values that depend on the environment. The NLoS probability is, then, equal to $1 - p_{u,v}^{\text{LoS}}$. $PL_{u,v}^{\text{LoS}}$ and $PL_{u,v}^{\text{NLoS}}$ are the PL for LoS and NLoS links and are, respectively, given in dB as follows:

$$PL_{u,v}^{\text{LoS}} = 10\nu \log_{10}\left(\frac{4\pi\sqrt{\text{Dist}(u,v)^2 + h_u^2}}{\lambda}\right) + \xi^{\text{LoS}}, \tag{8}$$

$$PL_{u,v}^{\text{NLoS}} = 10\nu \log_{10}\left(\frac{4\pi\sqrt{\text{Dist}(u,v)^2 + h_u^2}}{\lambda}\right) + \xi^{\text{NLoS}}, \tag{9}$$

where ν is the PL exponent, λ is the carrier wavelength, and ξ^{LoS} and ξ^{NLoS} are additional average losses for the LoS and NLoS links, respectively. Note here that higher UAV altitudes increase the chance of establishing a LoS link. However, it results in a larger distance between the transceiver. An adequate choice of h_u would reduce the PL effect and improve of the communication channel quality.

To ensure successful UAV operation, we consider that the received signal's power must be equal to or greater than the minimum required received signal strength denoted by P_r^{th}. Hence, the UAV has to be located at the range r_u^* such that the received power at a node placed at a distance r_u^* from (ϕ_u, ψ_u) is equal to P_r^{th}. In other words, for a given UAV altitude h_u, the following equation has to be solved:

$$\left[\frac{P_u^{tx}}{\sqrt{\overline{PL}_{u,v}}}\right]_{|d(u,v)=r_u^*} = P_r^{\text{th}}. \tag{10}$$

Using the above equation, we ensure that all the nodes located inside the disk centered in (ϕ_u, ψ_u) with the radius r_u^* detect the signal successfully when the UAV hovers at a 3D position (ϕ_u, ψ_u, h_u). Since the docking station is based on average statistics, the UAV operator should choose an UAV altitude h_u sufficiently high to guarantee a LoS probability close to 1 and avoid/minimize the mis-classification risk of PoIs. If equation (10) is not mathematically tractable, the range r_u can be determined via numerical methods such as the Newton method.

III. DOCKING STATION PLACEMENT

Before optimizing the scheduling procedure of the UAVs, we propose to determine permanent locations for the docking stations. This will represent a starting point for our next study in Section IV aiming at optimizing the UAV operations for ITS systems.

As the docking station will be permanently deployed, we propose to determine their locations based on the average statistics of the traffic in the geographical area and the minimum technical specifications of the UAVs. In other words, if the operator possesses UAVs with different battery capacities, denoted by $\bar{B}_d$ and speeds V_d, the worst case scenario will be considered. Hence, in the remaining part of this section, the index d related to the UAVs will be omitted.

A. PROBLEM FORMULATION

We associate to each PoI $n, n = 1 \ldots, N$, representing an intersection, a street, or a segment of road, a fitness value, denoted by f_n, measuring a metric evaluating the vehicle density, the event/incident frequency, or the vehicle speed, etc. The fitness can also be a weighted function of all these parameters. In Fig. 1, we present a realistic road network map of Qatar 2018 where major roads in the North West of Doha are illustrated. In the figure, each point has a different fitness value depending on the average traffic statistics. Some roads and intersections are characterized by higher fitness values due to, for example, high traffic jam or accident rates.

The objective of this section is to develop an optimization problem aiming at locating the docking stations in optimized positions such that the corresponding UAVs can cover the maximum number of PoIs with the highest fitness levels. Hence, the coverage efficiency (%) can be expressed

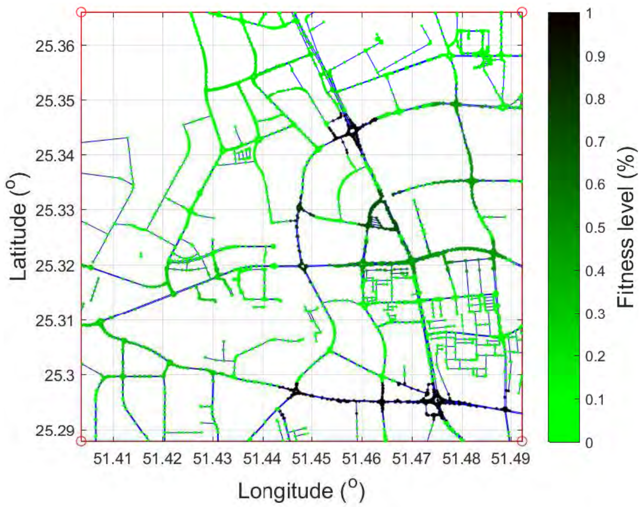


FIGURE 1. The traffic road network of the North West of Doha, Qatar (2018). The color bar indicates a metric representing certain statistics of the traffic in different intersections and road segments.

as follows:

$$\mathcal{C}_e^1 = \frac{\sum_{n=1}^N \epsilon_n f_n}{\sum_{n=1}^N f_n}, \tag{11}$$

where ϵ_n is a binary parameter indicating whether a point n can be covered by a UAV associated to one of the docking stations or not. If it is the case, $\epsilon_n = 1$. Otherwise, $\epsilon_n = 0$.

The placement of docking stations is not a straightforward task and depends on many factors. For instance, the docking stations must be installed in locations close enough to the N PoIs to ensure rapid deployment of UAVs in the case of emergency situations. We denote this time response related to a PoI n by $T^e(n), n = 1, \dots, N$. Hence, if the UAV requires a time less than or equal to $T^e(n)$ to reach a location (ϕ_u, ψ_u) distanced from (ϕ_n, ψ_n) by at maximum r_u^* then, the PoI n is assumed to be covered in a reasonable time. This time can differ from a PoI n to another according to the choice of the operator. Hence, the distance between a docking station s and a PoI n, $\text{Dist}(s, n)$, should not exceed the distance $d^{e*}(n)$ defined as follows:

$$d^{e*}(n) = T_e(n)V + r_u^*. \tag{12}$$

Moreover, the placement of docking stations should consider the UAV's battery level, which has not to be drained during the UAV mission to guarantee its safe execution. Hence, if the UAV's mission takes $T^{\text{cov}}(n)$ seconds for a given PoI n, then, the energy consumption of the UAV has to be less than the battery capacity as expressed in the following:

$$2P^{\text{mot}}T^{\text{fly}}(s, u) + \left(P^{\text{hov}} + P^{\text{s}} + P^{\text{com}}\right) T^{\text{cov}}(n) \leq \bar{B}, \tag{13}$$

where the first term in (13) accounts for the energy needed to guarantee a round trip flight between the docking station and a specific location (ϕ_u, ψ_u, h_u). The second term accounts for the energy consumption needed to cover the mission during $T^{\text{cov}}(n)$ seconds. Note that we consider that the UAV is fully loaded when it leaves the docking station. Consequently, the distance between a docking station s and the point n, $\text{Dist}(s, n) = \text{Dist}(s, u) + r_u^*$, has to respect the following condition:

$$\text{Dist}(s, n) \leq \underbrace{\left[\sqrt{\left(\frac{V\left(\bar{B} - \left(P^{\text{hov}} + P^{\text{s}} + P^{\text{com}}\right) T^{\text{mot}}(n)\right)}{2P^{\text{mot}}}\right)^2 - h_u^2} + r_u^*\right]^+}_{d^{b*}(n)}, \tag{14}$$

where $[.]^+$ indicates the max(., 0) function. In the sequel, we denote the right hand side of equation (14) by $d_b{}^*(n)$.[2]

It is worth to note that, for a fixed number of docking stations S, it will not be possible to cover all N PoIs mainly for large geographical area. Therefore, the UAV-ITS planner should target the PoIs with high fitness values f_n. To do so, we formulate the following optimization problem aiming at locating S docking stations in order to maximize the coverage efficiency of the UAV-based ITS:

$$\underset{(\phi_s, \psi_s), \forall s=1,\dots,S}{\text{maximize}} \ \mathcal{C}_e^2 = \frac{\sum_{n=1}^N \sum_{s=1}^S \epsilon_{sn} f_{sn}}{\sum_{n=1}^N f_n}, \tag{15}$$

$$\text{with } f_{sn} = \begin{cases} f_n, & \text{if } \text{Dist}(s, n) \leq \min(d^{e*}(n), d^{b*}(n)), \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{subject to: } \sum_{s=1}^S \epsilon_{sn} \leq 1, \quad \forall n = 1, \dots, N. \tag{16}$$

Constraint (16) is added to guarantee that the fitness f_n is only computed once, if a PoI n is simultaneously covered by more than one docking station. The value ϵ_{sn} is directly deduced from the locations of the docking stations. Hence, the selected docking station among the possible candidates is the closest one to the PoI n. In other words,

$$\epsilon_{s^*n} = 1, \text{ if } s^* = \underset{s=1,\dots,S \,/\, \text{Dist}(s,n) \leq \min(d^{e*}(n), d^{b*}(n))}{\arg\min} \text{Dist}(s, n). \tag{17}$$

Note that this optimization problem is developed to find the best locations (ϕ_s, ψ_s) for S UAV docking stations. Hence, it is designed for the case where a budget allowing the deployment of, at maximum, S docking stations in the area of interest is guaranteed. Another formulation might search for deploying the minimum number of docking stations to ensure a certain level of coverage efficiency. In this paper, we essentially focus our analysis on the optimization problem formulated in in (15). The algorithms that will be presented in the following section can be easily adapted to the other formulation.

[2] In the planning phase, it is assumed that each docking station has its own fleet and each UAV leaving that docking station is guaranteed to be able to return to it as indicated equation (14). This assumption is made to guarantee safe operation of the fleet in the case of isolated docking stations. Moreover, the planning phase is designed for the high-level operation of the system. Special cases and instantaneous operations of the system are taken into consideration in the scheduling process where, for each scenario, an activity plan for each UAV is determined. It indicates whether the UAV needs to go to another docking station or not according to the system requirements and event schedules.

B. PROPOSED ALGORITHMS FOR UAV DOCKING STATION PLACEMENT

In this section, we develop two algorithms determining permanent locations of the UAV docking stations in a geographical map given N PoIs defined by the ITS operator. The algorithms are designed using different conceptional constructions. In the first algorithm, we propose a modified version of the K-means clustering method which originally partitions all the N PoIs into K clusters [36]. In the second algorithm, we utilize the bio-inspired PSO algorithm [37].

1) PENALIZED AND WEIGHTED K-MEANS ALGORITHM

The PWkMeans is a modified version of the traditional K-means algorithm, which designed to assign the points (or data) to the closest centroid. In the modified version, we propose to penalize the assignment since we do not aim to associate all the PoIs to the docking stations as the clustering operation is conditioned by the distance given in (15). Indeed, some PoIs may not be assigned to any docking station. Moreover, the modified version of the algorithm, we add a weight to each PoI n reflecting its fitness value f_n which differs from the ones of the other PoIs. Hence, higher weights are given to PoIs with higher fitness values so that the clustering algorithm gives priority to those PoIs during the assignment process and thus, ensure the maximization of the coverage efficiency. In this context, we proceed with observation weights rather than feature weights.

The first step of the PWkMeans is to initialize the locations of the S docking stations. The initial locations can be randomized or pre-defined by the operator. This initialization step has an important impact on the final algorithm results. Therefore, we opt to adopt the initialization step of the K-means++ approach presented in [36] to select the initial centroids and hence, enhance the convergence efficiency. Afterwards, the algorithm associates the N PoIs to the centroids based on the distances $\text{Dist}(s, n)$ by assigning high weights to the PoIs with high fitnesses f_n and vice versa. This can be done by replicating the PoIs according to their fitness values following a particular scale. For instance, a PoI with a fitness close to 1 will be replicated 10 times in the data set while a PoI with a fitness close to 0 will not be replicated. Then, new centroids can be obtained by averaging over the coordinates of the PoIs associated to each old centroid that respects the distance condition given in (15). These steps are repeated until no changes are made, i.e., no further change in the centroid locations. The PWkMeans algorithm is given in Algorithm 1. At each execution, the PWkMeans may converge to a different local optimum. Hence, it is recommended to execute it multiple times to select the combination providing the highest coverage efficiency.

Algorithm 1 PWkMeans for Docking Station Placement

$t = 0$.
Generate initial random locations $(\phi_s(t), \psi_s(t))$ for S docking stations using the k-means++ method.
Generate $\tilde{N}$ points from the N PoIs by replicating them according to a given scale.
while Not converged **do**
  For each centroid $(\phi_s(t), \psi_s(t))$, find the closest points from $\tilde{N}$.
  $t = t + 1$.
  Find the new locations of $(\phi_s(t), \psi_s(t))$ by averaging over the coordinates of the associated points that respect the distance condition.
end while
Return $(\phi_s^*, \psi_s^*) = (\phi_s(t), \psi_s(t))$.

2) PARTICLE SWARM OPTIMIZATION

Evolutionary algorithms can be also employed to optimize the UAV docking station placement. In this paper, we propose to use PSO, which is characterized by the following advantages compared to other evolutionary methods: The search process in PSO is simple easy to implement, it requires the manipulation of few numerical parameters (e.g., the number of particles and acceleration factors), and it needs a low computational cost when executed with a small number of particles [38].

Let us denote by $\boldsymbol{C}$ the matrix of size $2 \times S$ containing the coordinates of S docking station as follows:

$$\boldsymbol{C} = \begin{bmatrix} \phi_1 & \phi_2 & \dots & \phi_S \\ \psi_1 & \psi_2 & \dots & \psi_S \end{bmatrix}. \tag{18}$$

The goal is to find the best combination of $\boldsymbol{C}$ such that the coverage efficiency is maximized. The first step in PSO is to create B particles composed of randomly generated docking station coordinates $\boldsymbol{C}(0, b)$, $b = 1, \dots, B$ to form an initial population. Then, at each iteration i and for each particle b, it associates to each docking station located at $(\phi_s(i, b), \psi_s(i, b))$ the PoIs that respect the distance condition given in (15). Then, it computes the coverage efficiencies $\mathcal{C}_e^2(i, b)$ associated to each particle b. Among all these particles, it records the one providing the highest coverage efficiency, which is named the global particle for this iteration and denoted by $\boldsymbol{C}^{\text{G}}$. In addition, for each particle b, it maintains a record of its best performance through the iterations known as the local position of particle b and denoted by $\boldsymbol{C}^{\text{L}}(b)$. Afterwards, it computes a velocity term $\boldsymbol{V}(i, b)$ as follows:

$$\boldsymbol{V}(i, b) = \xi \boldsymbol{V}(i-1, b) + \boldsymbol{\Psi}_i \left(\boldsymbol{C}^{\text{L}}(b) - \boldsymbol{C}(i, b) \right) + \boldsymbol{\Phi}_i \left(\boldsymbol{C}^{\text{G}} - \boldsymbol{C}(i, b) \right), \tag{19}$$

where ξ is the inertia weight and $\boldsymbol{\Psi}_i$ and $\boldsymbol{\Phi}_i$ are two matrices whose elements are uniformly generated in $[0, 2]$ at each iteration i. The velocity term is used to shift the locations of each particle b based on its best local position as well as the global particle. Finally, PSO updates each particle $\boldsymbol{C}(i, b)$ as follows:

$$\boldsymbol{C}(i, b) = \boldsymbol{C}(i, b) + \boldsymbol{v}(i, b). \tag{20}$$

Algorithm 2 PSO for Docking Station Placement

$i = 0$.
Generate an initial population composed of B random particles $\boldsymbol{C}(i, b)$.
while Not converged **do**
  for $b = 1, \cdots, B$ **do**
    Associate to each docking station s of particle b the closest PoIs that respect the distance condition given (15).
    Compute the coverage efficiency of particle b: $\mathcal{C}_e^2(i, b)$.
  end for
  Find the global particle $\boldsymbol{C}^{\mathrm{G}}$ and the local position $\boldsymbol{C}^{\mathrm{L}}(b)$ for each particle b.
  Adjust the velocities and positions of all particles using equations (19) and (20).
  $i = i + 1$.
end while

This process is repeated until reaching convergence either by attaining the maximum number of iterations or by stopping the algorithm when the achieved coverage efficiency remains constant after a certain number of iterations. Algorithm 2 provides a pseudo-code of the placement of UAV docking stations using PSO.

It is worth to note that, in the case where RSUs are already installed in the area of interest, the placement of docking stations in locations close to RSUs will be automatically avoided by the algorithms to maximize the coverage efficiency as the surrounding PoIs are already covered by RSUs and hence, their corresponding ϵ_n are already set to 1.

IV. UAV SCHEDULING

In this section, we develop a scheduling framework to manage a fleet of UAVs. We assume that D UAVs are initially placed at the docking stations deployed using the approach presented in Section III. The objective is to exploit these UAVs to cover spatially and temporally distributed events occurring in the area of interest over a long period of time. The proactive approach aims to employ these UAVs in sequential and parallel manners such that the events are effectively covered and the handoffs between UAVs are smoothly performed while respecting the energy availability at each UAV. The UAVs can be used to successively cover multiple events and must guarantee sufficient energy to return to the docking station so they can replenish their batteries. The investigated framework presents some resemblance to the parallel machine scheduling problem where machines are assigned to execute multiple jobs [39], [40]. However, in our framework, we are taking into account more complex features related to the machines, i.e., in our case the UAVs, that have limited battery lifetime and require to regularly return to the one of the docking stations to recharge their batteries. Moreover, the delay due to the motion of the UAVs need to be considered to ensure effective coverage. This makes the optimization of the UAV management more challenging and must be designed to guarantee the smooth operations of the UAVs.

A. EVENTS AND UAV ACTIONS

We assume that the time horizon T is divided into K time slots of length T_k and the E events are pre-scheduled in the area of interest. Each event is defined by its geographical location (ϕ_e, ψ_e, h_e), its starting time τ_e, and its duration Δ_e where $e = 1, \ldots, E$. These events are pre-planned by the operator. They can correspond to various ITS applications involving UAVs such as regular monitoring of intersections and important rods during peak hours, periodic data collection missions from RSUs, or acting as flying speed cameras, etc.

Initially, the D UAVs are placed in docking stations located at the optimized locations $(\phi_s, \psi_s, 0)$ where they can charge their batteries with a charging power denoted by P^{ch}. Hence, in total, there are $E + S$ possible locations for UAVs' deployment. In the sequel, we identify the docking station s as event $E + s$.

In the UAV scheduling framework, we aim to manage the UAV fleet by determining what is the action that is performed by a UAV at each instant. Four actions are possible:

- Covering an event: The UAV is located at one of the events e, where $e = 1, \ldots, E$ to execute a particular mission. When executing this mission, it is assumed that the UAV is hovering at a static location and consuming, in total, the following power level $P^{\mathrm{hov}} + P^s + P_e^{\mathrm{com}}$. Notice that the index u is replaced by the index e in P_e^{com} since the power level related to the communication service depends on the location of the event chosen by the operator.
- Charging its battery: The UAV is located at one of the docking stations $e = E + 1, \ldots, E + S$ to charge its battery with a charging power P^{ch}.
- Flying from an event to another: The UAV will fly from an event e' to another event e where $e', e \in \{1, \ldots, E + S\}$. During its motion, a UAV d consumes an amount of power equal to $P_d^{\mathrm{mot}} = P^{\mathrm{hov}} + P_d^{\mathrm{har}}$. This also includes the UAV trip from a docking station to another.
- Waiting at an event: The UAV is located at an event e waiting to do an action. This case can happen when the UAV is waiting at a docking station without charging its battery because its full, in this case, the UAV does not consume any power, or waiting at one of the events, $e = 1, \ldots, E$ until it starts to cover it and in this case, it consumes $P^{\mathrm{hov}} + P^s$.

The UAVs will need to switch between these different actions that have different time duration depending on the spatio-temporal distributions of the events as well as the locations of the docking stations.

B. DECISION VARIABLES AND CONSTRAINTS

The goal of this section is to define the different decision variables and formulate the necessary constraints to ensure parallel and sequential scheduling of the fleet of UAVs with minimum energy consumption while guaranteeing sustained coverage of the events.

TABLE 1. List of decision variables.

Decision Variable	Type	Size	Meaning
$c(d,e,k)$	binary	$D\cdot(E+S)\cdot K$	equals 1 if UAV d is serving event e during time slot k,
$\Delta c(d,e,k)$	non-negative real	$D\cdot(E+S)\cdot K$	time spent by UAV d to cover event e during time slot k
$w(d,e,k)$	binary	$D\cdot(E+S)\cdot K$	equals 1 if UAV d is waiting at event e during time slot k
$\Delta w(d,e,k)$	non-negative real	$D\cdot(E+S)\cdot K$	waiting time of UAV d at event e during time slot k
$\Delta f(e',e,d,k)$	non-negative real	$(E+S)\cdot(E+S-1)\cdot D\cdot K$	flying time of UAV d to move from event e' to event e during time slot k
$i(d,d',e,k)$	binary	$D\cdot\binom{D-1}{2}\cdot E\cdot K$	equals 1 if UAV d and UAV d' can sequentially cover event e during time slot k

In Table 1, we list the primary decision variables of the problem. Note that:

• When $c(d, e, k) = 1$ where $e > E$ it means that UAV d is charging its battery during time slot k at docking station s (i.e., $s = E - e$). The charging period is determined by the decision variable $\Delta c(d, e, k)$.

• It is possible that a UAV is located at the docking station s but not charging its battery, in this case, $c(d, E+s, k) = 0$ and $w(d, E + s, k) = 1$. The waiting time of the UAV is defined by $\Delta w(d, e, k)$.

• The size of $\Delta f(e', e, d, k)$ is reduced as we eliminate the cases when $e' = e$.

• The decision variable $i(d, d', e, k)$ is introduced to indicate whether two UAVs are sequentially covering the same event during the same time slot or not and hence, redundant use of UAVs and collision are avoided. Its size is reduced since $i(d, d', e, k) = i(d', d, e, k)$ for $d \neq d'$ and the case $d = d'$ is ignored.

In the following, we present all constraints required for safe operation of the UAVs.

1) EVENT COVERAGE CONSTRAINT

The following constraint are related to the binary variable $c(d, e, k)$:

$$s_{d,e,k} \leq \delta_{e,k}, \quad \forall d = 1, \ldots, D, \forall e = 1, \ldots, E, \quad \forall k = 1, \ldots, K, \tag{21}$$

$$\sum_{e=1}^{E+S} s_{d,e,k} \leq 1, \quad \forall d = 1, \ldots, D, \forall k = 1, \ldots, K. \tag{22}$$

Constraint (21) prohibits the optimizer to assign a UAV d to an event e during time slot k if the event does not occurs during that time slot. The parameter $\delta_{e,k}$ is introduced to indicate whether an event e is occurring during time slot k or not; (($\delta_{e,k} = 1$ if it is the case). Constraint (22) avoids the parallel use of one UAV to cover multiple events. Hence, a UAV can cover only one event during each time slot k. Notice that it is not mandatory that a UAV will cover an event e during the whole period T_k. Indeed, the UAV may spend some of the time to cover the event e while, during the rest of the time slot, it may be in motion to another event, for instance, to the docking station due to a lack of energy.

Multiple UAVs can be sequentially used to cover a single event. Hence, their total coverage time should not exceed the duration of the event. It may not be possible to completely cover all the events and some events might be missed due to limited number of UAVs and/or lack of energy. Hence, the operator can define a certain target to be achieved for each event or for all events together. This depends on the available resources provided by the operator. For the case of local tolerance, i.e., associated to each event, the constraint can be written as follows:

$$\kappa_e \Delta_e \leq \sum_{d=1}^{D} \sum_{k=1}^{K} c(d, e, k) \Delta c(d, e, k) \leq \Delta_e, \quad \forall e = 1, \ldots, E. \tag{23}$$

where κ_e is the tolerance parameter associated to each event e, ($0 \leq \kappa_e \leq 1$). If it is about a global tolerance then, the coverage efficiency constraint can be written as:

$$\sum_{d=1}^{D} \sum_{k=1}^{K} c(d, e, k) \Delta c(d, e, k) \leq \Delta_e, \quad \forall e = 1, \ldots, E, \tag{24a}$$

and

$$\sum_{k=1}^{K} \sum_{e=1}^{E} \sum_{d=1}^{D} c(d, e, k) \Delta c(d, e, k) \geq \bar{\mathcal{C}}, \tag{24b}$$

where $\mathcal{C}$ is the minimum target coverage of the operator. Note that staying at any of the docking stations is not limited in time. Therefore, constraints (23) and (24) are not imposed for the events $E + s$, where $s = 1, \ldots, S$.

2) WAITING CONSTRAINTS

It might be possible that a UAV will need to statically hover at a single location to wait for an event to start. In this work, we assume that, if they need to wait, UAVs must first move to the upcoming event and wait at its location. It is redundant that a UAV d remains waiting at an event which has already expired. Hence the following constraints are imposed:

$$w(d, e, k) \leq \sum_{k'=k+1}^{K} \delta_{e,k}, \quad \forall e = 1, \ldots, E, \quad \forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K \tag{25}$$

$$\sum_{e=1}^{E+S} w(d, e, k) \leq 1, \quad \forall d = 1, \ldots, D, \ \forall k = 1, \ldots, K. \tag{26}$$

Constraint (26) ensures that a UAV can wait at maximum at one event during each time slot. Recall that UAVs have the freedom to wait at the docking stations.

3) FLYING CONSTRAINT

If a UAV d decides to fly from event e' to event e, its trip time, which might be summed over multiple time slots, has to be exactly equal to $T_d^{\text{fly}}(e', e)$ defined in (3). Given the fact that one UAV can be only located at one event during a time slot and assuming that a UAV d leaves event e' during time slot k' and reaches event e during time slot k then, the following constraint has to be maintained:

$$\sum_{j=k'}^{k} \Delta f(e', e, d, j) = \begin{cases} T_d^{\text{fly}}(e', e) & \text{if UAV } d \text{ is flying from } e' \text{ to } e, \\ 0 & \text{otherwise,} \end{cases}$$
$$\forall d = 1, \ldots, D, \quad \forall e', e \in \{1, \ldots, E+S\} \text{ where } e' \neq e, \quad \forall k', k \in \{1, \ldots, K\} \text{ where } k' \leq k. \quad (27)$$

4) SEQUENTIAL COVERAGE CONSTRAINT

To avoid the possibility that two or more UAVs simultaneously cover the same event, we propose to introduce the sequential coverage constraint interlinking the decision variables $c(d, e, k)$ and $i(d, d', e, k)$. However, let first introduce a variable denoted by $\Delta L(d, e, k)$ that represents the duration limited between the instant at which time slot k starts and the instant at which UAV d starts covering event e. For instance, if the UAV d starts covering the event e at the beginning of time slot k then, $\Delta L(d, e, k) = 0$. If the UAV starts covering the event e after a certain delay due to its travel and/or waiting times then, $\Delta L(d, e, k) = \Delta f(e', e, d, k) + w(d, e, k)\Delta w(d, e, k)$ assuming that e' is the previous location of UAV d. Finally, if a UAV d is not covering the event e at all during time slot k, $\Delta L(d, e, k) = T_k$. A general expression of $\Delta L_{d,e,k}$ is given as follows:

$$\begin{aligned} &\Delta L(d, e, k) \\ &= \delta_{e,k}\left(\sum_{e'=1}^{E+S} \Delta f(e', e, d, k) + w(d, e, k)\Delta w(d, e, k)\right) \\ &\quad + (1 - c(d, e, k))\, T_k, \ \forall d = 1, \ldots, D, \\ &\qquad \forall e = 1, \ldots, E, \forall k = 1, \ldots, K. \end{aligned} \quad (28)$$

Notice that the variables $\Delta L(d, e, k)$ are not considered as decision variables as they can be expressed as a function of the other primary decision variables.

The following condition is applied for two UAVs covering the same event to guarantee its sequential coverage:

$$\begin{aligned} &\text{If } \Delta L(d, e, k) \geq \Delta L(d', e, k) + c(d', e, k)\Delta c(d', e, k) \\ &\quad \text{or } \Delta L(d', e, k) \geq \Delta L(d, e, k) + c(d, e, k)\Delta c(d, e, k) \text{ then,} \\ &\quad i(d, d', e, k) = 1, \ \forall d, d' \in \{1, \ldots, D\}, \forall e = 1, \ldots, E, \\ &\qquad \forall k = 1, \ldots, K. \end{aligned} \quad (29)$$

Therefore, a UAV d can serve an event e during a time slot k only if it is able to satisfy constraint (29) with all the other UAVs. Hence, the binary variable $c(d, e, k)$ is conditioned upon the values of $i(d, d', e, k)$ as follows:

$$c(d, e, k) \leq \sum_{d=1}^{D} \sum_{d'=d+1}^{D} i(d, d', e, k) - \left(\frac{D(D-1)}{2} - 1\right),$$
$$\forall d = 1, \ldots, D, \forall e = 1, \ldots, E, \forall k = 1, \ldots, K. \quad (30)$$

Hence, all events will be covered in a sequential manner. Indeed, a UAV will be sent to an event only if it is ensured that it will not overlap with other UAVs. Notice that constraint (30) is not applicable for event $E + s$, $s = 1, \ldots, S$ as the UAVs can be co-located in the docking stations.

5) ENERGY CONSUMPTION CONDITION

We first define the following amounts of energy consumed or charged by each UAV d until time slot k:

- Total energy consumption to cover events denoted by $E_{d,k}^{\text{ser}}$:

$$E_{d,k}^{\text{ser}} = \sum_{j=1}^{k} \left(P_d^{\text{hov}} + P_d^{\text{ser}}\right) \sum_{e=1}^{E} s(d, e, j)\Delta s(d, e, j),$$
$$\forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K. \quad (31)$$

- Total energy consumption due to waiting denoted by $E_{d,k}^{\text{wait}}$:

$$E_{d,k}^{\text{wait}} = \sum_{j=1}^{k} P_d^{\text{hov}} \sum_{e=1}^{E} w(d, e, j)\Delta w(d, e, j)\ ,$$
$$\forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K. \quad (32)$$

- Total energy consumption due to motion denoted by $E_{d,k}^{\text{fly}}$:

$$E_{d,k}^{\text{fly}} = \sum_{j=1}^{k} \left(P_d^{\text{hov}} + P_d^{\text{tr}}\right) \sum_{\substack{e=1 \\ e \neq e'}}^{E+S} \sum_{e'=1}^{E+S} \Delta f(e', e, d, j)\,,$$
$$\forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K. \quad (33)$$

- Total energy charged denoted by $E_{d,k}^{\text{char}}$:

$$E_{d,k}^{\text{char}} = \sum_{j=1}^{k} P^{\text{ch}} \sum_{s=1}^{S} s(d, E+s, j)\Delta s(d, E+s, j)\,,$$
$$\forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K. \quad (34)$$

Hence, in terms of energy consumption and at each time slot k, a UAV d cannot consume an amount of energy higher than the amount of energy stored in its battery as given below:

$$E_{d,k}^{\text{ser}} + E_{d,k}^{\text{wait}} + E_{d,k}^{\text{fly}} \leq B_d^0 + E_{d,k-1}^{\text{char}},$$
$$\forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K, \quad (35)$$

where B_d^0 is the initial amount of energy stored at the battery of UAV d. Constraints (35) guarantee that the amounts of consumed energy by UAV d until time slot k cannot exceed the amount of energy stored before that period, i.e., the initial battery level plus the amounts of energy acquired from the docking station source during the previous time slots. Notice that, for $k = 1$, we consider that $E_{d,0}^{\text{char}} = 0$. Finally, we denote by E_d the total energy consumption of a UAV d which is equal to the sum of the energies consumed over all time slots and expressed as:

$$E_d = E_{d,K}^{\text{ser}} + E_{d,K}^{\text{wait}} + E_{d,K}^{\text{fly}}, \ \forall d = 1, \ldots, D. \quad (36)$$

6) BATTERY CAPACITY CONSTRAINT

On the other hand, we need to ensure that, when recharging, the UAV battery capacity limit $\bar{B}_d$ is not violated and this during each time slot:

$$B_d^0 + E_{d,k}^{\text{char}} - \left(E_{d,k}^{\text{ser}} + E_{d,k}^{\text{wait}} + E_{d,k}^{\text{fly}}\right) \leq \bar{B}_d, \quad \forall d = 1, \ldots, D, \forall k = 1, \ldots, K. \quad (37)$$

In order to not exceed the battery capacity, a UAV can land on one ofthe docking stations without necessary charging its battery, i.e., $w(d, E+s, k) = 1$ and $s(d, E+s, k) = 0$.

7) TIME INTERVAL CONSTRAINT

This constraint guarantees the cohesion of the time scheduling such that the total spent by a UAV d to move from an event to another and/or to cover and wait at an event must be equal to the time slot length as follows:

$$\sum_{e=1}^{E+S} c(d, e, k)\Delta c(d, e, k) + \sum_{e=1}^{E+S} w(d, e, k)\Delta w(d, e, k) + \sum_{\substack{e=1 \\ e \neq e'}}^{E+S} \sum_{e'=1}^{E+S} \Delta f(e', e, d, k) = T_k, \quad \forall d = 1, \ldots, D, \quad \forall k = 1, \ldots, K. \quad (38)$$

C. OBJECTIVE FUNCTION AND OPTIMIZATION PROBLEM

The objective of the framework is to minimize the total energy consumption while meeting the coverage efficiency targeted by the operator. The objective function corresponds then to the sum of the energy consumption over all UAVs. The UAV management optimization problem is written as follows:

$$\underset{\substack{c, \Delta c, w, \Delta w \\ \Delta f, i}}{\text{minimize}} \ \mathcal{F} = \sum_{d=1}^{D} \mathcal{E}_d, \quad (39)$$

subject to (21), (22), (23) or (24), (25), (26), (27), (29), (30), (35), (37), and (38),

Initialization of the UAVs' locations constraints. (40)

The parameters $\boldsymbol{c}$, $\boldsymbol{\Delta c}$, $\boldsymbol{w}$, $\boldsymbol{\Delta w}$, $\boldsymbol{\Delta f}$, and $\boldsymbol{i}$ are the set containing the decision variables $c(d, e, k)$, $\boldsymbol{\Delta c}(d, e, k)$, $\boldsymbol{w}(d, e, k)$, $\boldsymbol{\Delta w}(d, e, k)$, $\boldsymbol{\Delta f}(e', e, d, k)$, and $\boldsymbol{i}(d, d', e, k)$, respectively. It should be noted that, in addition to the already discussed constraints, the initial locations of each UAV must initialized (i.e., (40)). Usually, the UAVs are initially located at the docking stations. Each can be waiting and/or charging at one of the docking stations. The operator of the UAVs decides where to deploy each UAV at the beginning of the time horizon. During their operations, it is possible that the UAVs will go to different docking stations to charge their batteries as long it is needed and possible. The problem formulation does not impose that each UAV is associated to a unique docking station. However, thanks to the flexibility of the UAVs, even if they are initially located at a central docking station, they can easily adjust their locations according to the system need especially if we consider continuous operation of the system. For instance, at the beginning of each cycle, the UAV will start at the central docking station and according to the scheduled missions, the optimizer will automatically determine where each UAV needs to land and recharge its battery, and the end of the cycle, the UAVs return to the central docking station. Optimizing the initial number of UAVs per docking stations would help in minimizing the energy consumption.

The outputs of the problem include the starting times of each of the UAV actions presented earlier and their corresponding durations for each UAV in addition to the location of each UAV at each time slot k where $k = 1, \ldots, K$.

Notice that the above optimization problem is non-linear due to the non-linearity of the objective function and some of the constraints, e.g., products of decision variables, and existing logical conditions, e.g., logical *OR*. Hence, it is formulated a MINLP problem that cannot optimally solved. However, it is possible to convert it to a linear one by adopting some linearization techniques such as the use of multiple linear constraints to model the products of decision variables, the big-M linearization technique, and the introduction of some slack variables to linearize some of the constraints. More details about the linearize techniques that are adopted can be found in [28].

Once the problem is converted to a MILP problem, its optimal solution can be determined using well-known techniques, namely the branch-and-bound algorithm implemented in off-the-shelf software, e.g., CPLEX, CVX/Gurobi [41]–[43]. However, the MILP is a non-deterministic polynomial time (NP-Hard) problem requiring a high computational time to converge especially for large scale problem. Traditional heuristic approaches such as evolutionary algorithms may be used to achieve sub-optimal solutions after certain problem relaxation. Nevertheless, it is worth to note that the optimization problem is designed for a proactive scheduling and will be solved once for each period of time. For large scale scenarios, the UAV operator can divide the time horizon into multiple sub-periods following the temporal distribution of the events. As an example, the events can be divided into multiple groups of events as they are sufficiently spaced from each others. In case where the docking stations are sufficiently far from each other, the scheduling problem could be applied based on their locations. For example, the schedules of the UAVs associated to docking station A is determined independently of the schedules of the UAVs associated to docking stations B and C, located near each other.

In this paper, we have focused on the event category where a MILP is solved to jointly find the schedules of all UAVs during the whole timespan. In case of emergency situations requiring rapid deployment of the UAVs, real-time decision-making approaches are needed. Nevertheless, theoretically, the proposed optimal solution can still be implemented with for real-time situations where the MILP is re-executed when an unexpected event occur. The MILP will be then executed for the remaining timespan while considering the new unexpected event. At that moment, the starting time and the location of the unexpected event are known and a fixed duration can be given to the event, e.g. 20 minutes so the UAV can have sufficient time to gather necessary information. However, this case is not really practical because of the high complexity of the MILP, which makes it inefficient to determine a new schedule for the fleet to take prompt actions. In our future work, we will focus on designing rapid solutions to deal with unexpected events and emergency situations.

Another remark regarding the optimization problem is that it is feasible if many parameters such as the battery capacity $\bar{B}_d$, the duration of the events p_e as well as their spatial and temporal occurrence, the different power levels of the UAVs P^{hov}, P^{har}, and P_d^{com}, and the charging power level of the docking stations are selected in a reasonable and homogenous manner. For example, the used UAVs are chosen such that their batteries are designed for sufficiently long travel time so they can be employed in large geographical areas. Moreover, the number of deployed UAVs D and the tolerance parameters κ_e or $\bar{\mathcal{C}}$ must be well-chosen.

It is worth to note that the proposed formulation can be implemented with any type of discretization of the time horizon. In other words, the time horizon can be into K time slots with uniform lengths or with different/dynamic lengths. In fact, the proposed approach enables continuous decision making where the statuses of the UAVs can be exactly updated at the beginning, at the end, or between these two limits of each time slot. Hence, unlike the traditional time-indexed formulations, which impose that the status of the UAV must be the same during the duration of the time slot, the proposed formulation tolerates the change of the UAV statuses during the time slot. As it will be shown in the simulation result section, during the same time slot, a UAV can sequentially fly, wait, and then cover an event.

V. SELECTED NUMERICAL RESULTS

In this section, we provide some numerical results for the docking station placement and the UAV scheduling methods for different scenarios applied to the map of Doha city, Qatar given in Fig. 1. We start by introducing the system parameters used in our simulations then, investigates the outputs of the PSO and PWkMeans algorithms. Afterwards, we study the optimal scheduling of UAVs and their behaviors for different scenarios.

A. SIMULATION PARAMETERS

We use the free editable map provided by the OpenStreetMap project to obtain the road network data of the city of Doha, Qatar [44]. We focus our analysis on the ''trunk'', ''motorway'', ''primary'', ''secondary'' and ''tertiary'' road types as well as their associated links and we ignore the special road types such as ''footway'', ''residential'', and ''living_street'', etc. We use as UAV battery the zippy flight max lithium polymer battery[3] with a storage capacity of 27 Wh. The remaining UAV and system parameters are given in Table 2. Based on these parameters, the UAV range is computed to

[3]http://UAVsarefun.com/BatteriesForUAV.html

TABLE 2. UAV and system parameters.

Parameter	Value	Parameter	Value	Parameter	Value
m_{tot} (kg)	1.5	r_p (cm)	20	n_p	4
P_{full} (W)	5	P_s (W)	0.1	$v_{\max}$ (m/s)	28
v_u (m/s)	18	γ_u	4	δ_u (W)	6.8
P_u^{tx} (dBm)	-5	$P_{\min}$ (dBm)	-90	λ (m)	0.125
α	9.6	β	0.29	ξ_{LoS} (dB)	1
ξ_{NLoS} (dB)	8	ν	2	B (kJ)	399.6

be approximately equal to $r_u^* \approx 121$ m for a UAV altitude $h_u = 60$ m. The fitness metric used in the simulations is obtained from [45] where f_n is given as:

$$f_n = \eta_1 \rho_n^{\text{AC}} + \eta_2 \rho_n^{\text{AV}} + \eta_3 \rho_n^{\text{VH}}, \tag{41}$$

where ρ_n^{AC}, ρ_n^{AV}, and ρ_n^{VH} are metrics reflecting the accident frequency, average flow speed, and density of vehicles at the PoI n. The factors $\eta_, i$, $i = 1, 2, 3$ are coefficients associated to each metric. For simplification, in Fig. 1, the fitness level are represented by colors where dark colors indicate high fitness level. Finally, PSO and PWkMeans are executed for 20 times in order to find the best placement of docking stations. The PSO is run for $B = 12$ particles and interrupted if the coverage efficiency is unchanged for 20 consecutive iterations.

Since placing docking stations is permanent, we have used average statistics of the road network for this task. However, to deal with the varied behavior of the road network, the scheduling is performed according to the traffic variation and the zones to monitor which can vary during the day/makespan. As it will be shown in the following results, the UAV will move from a docking station to another depending on the needs of the sub-areas.

B. RESULTS FOR THE UAV DOCKING STATION PLACEMENT

In Figs. 2(a)-2(d), we depict the locations of the docking stations using the PWkMeans and PSO algorithms for different number of stations ($S = 4$ and $S = 8$). In Figs. 2(a)-2(b), four docking stations are deployed. Both algorithms cover almost the same regions characterized with high fitness levels as indicated in Fig. 1 (Red regions). The obtained docking station locations are very near to each other which results in a very close coverage efficiency levels (66.2% and 64.2% for PSO and PWkMeans, respectively). Deploying a higher number of docking stations ($S = 8$) as given in Figs. 2(c)-2(d) increases the coverage efficiency levels to 86.54% and 84%, respectively. However, the obtained locations are different for some of the docking stations. This is due to the fact that the number of docking stations is not enough to cover all of the PoIs and hence, different local solutions can be obtained.

In Fig. 3, we plot the achieved coverage efficiency with respect to the minimum required received signal strength P_r^{th} in order to enable safe communication by the UAV and a ground node for $S = 2$ and $S = 4$. This scenario can corresponds to the case of a UAV acting as a flying RSU communicating with ground vehicles. In this figure, we have compared the performance of our algorithms with the ones of a deterministic greedy approach. The greedy algorithm places

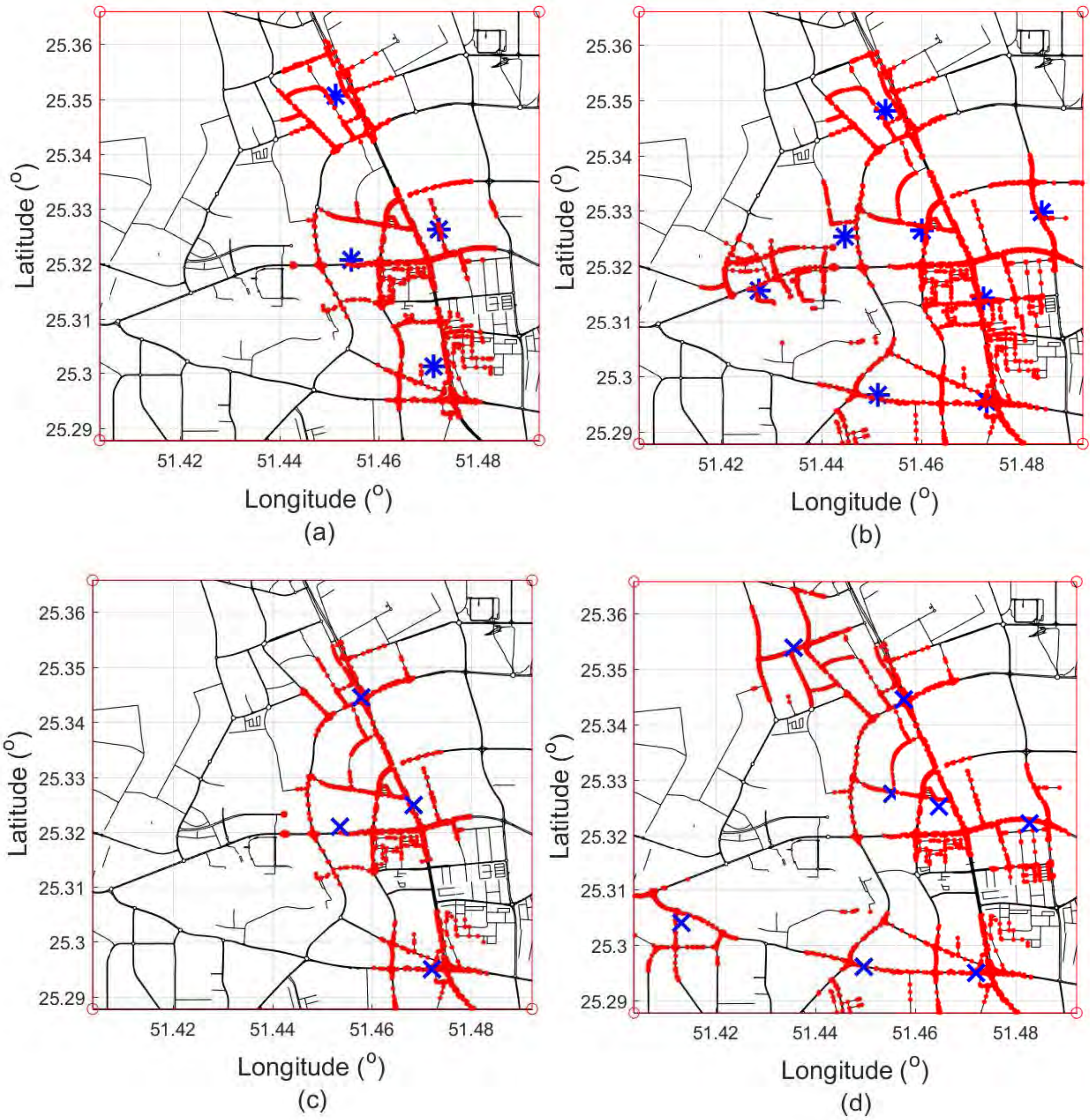


FIGURE 2. Placement of docking stations (a) PSO $S = 4$, (b) PSO $S = 8$, (c) PWkMeans $S = 4$, and (d) PWkMeans $S = 8$.

a docking station at each iteration. It scans the map to test pre-defined locations forming a grid. The docking station is placed at the location providing the highest coverage efficiency. The corresponding covered PoIs are then eliminated for the next iteration, where the algorithm looks for placing the next docking station. The algorithm converges when all the docking stations are placed.

In general, the figure shows that as the P_r^{th} increases, the coverage efficiency decreases. This is due to the fact that the UAV's range is shrunk and hence, the docking stations need to be installed in locations closer to important PoIs. This results in reducing the chance of covering other nearby PoIs, which confirms that the required QoS level constitutes an important parameter in the docking station placement decision. It is also important to note that maximizing the coverage efficiency is not equivalent to the maximization of the number of covered PoIs. The proposed algorithms target the PoIs having higher fitness first. Then, remaining docking stations are deployed to cover other less important regions having lower fitness values, in other words, regions with less incidents and accidents rate. Finally, Fig. 3 shows that the PSO algorithm always outperforms the PWkMeans clustering algorithm and the greedy one with a gap not exceeding 10%. The PWkmeans and greedy algorithms achieve close performance; the former algorithm efficiency is very dependent on its initialization step, while the greedy algorithm depends on its browsing grid size.

Fig. 4 depicts the case of placing four ($S = 4$) UAV docking station placement in the presence of deployed RSUs. The RSUs are assumed to be installed along a road segment located at the south of the map (green-colored). The PSO algorithm places three docking stations at the same locations

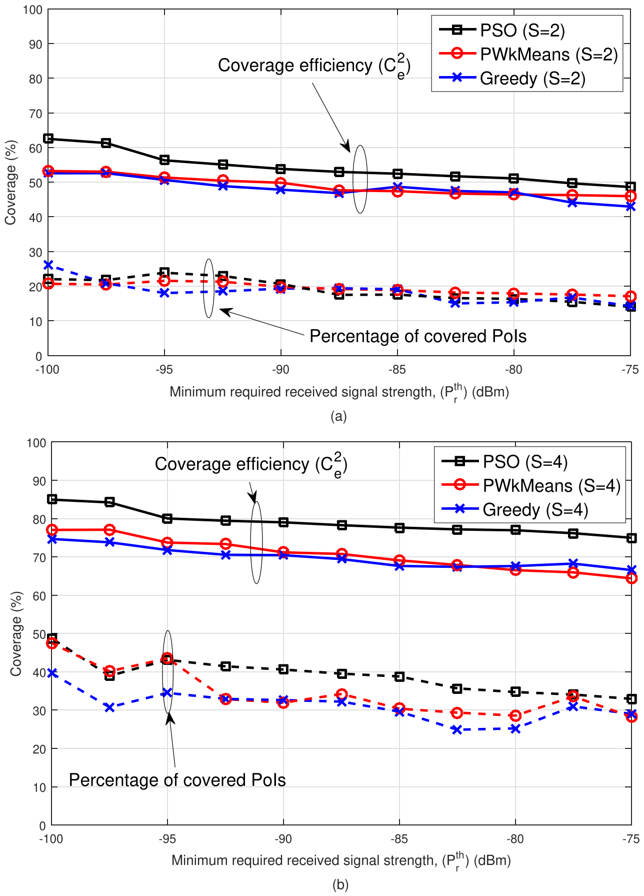


FIGURE 3. Coverage efficiency versus minimum required received signal strength using PSO, PWkMeans, and a greedy algorithm with (a) $S = 2$ and (b) $S = 4$.

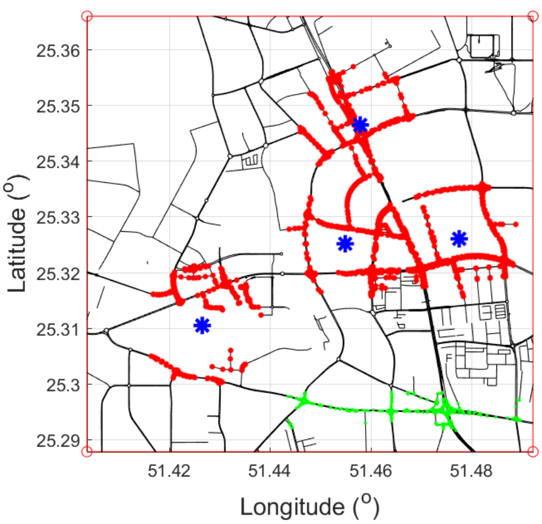


FIGURE 4. Placement of $S = 4$ docking stations in the presence of RSUs using PSO algorithm.

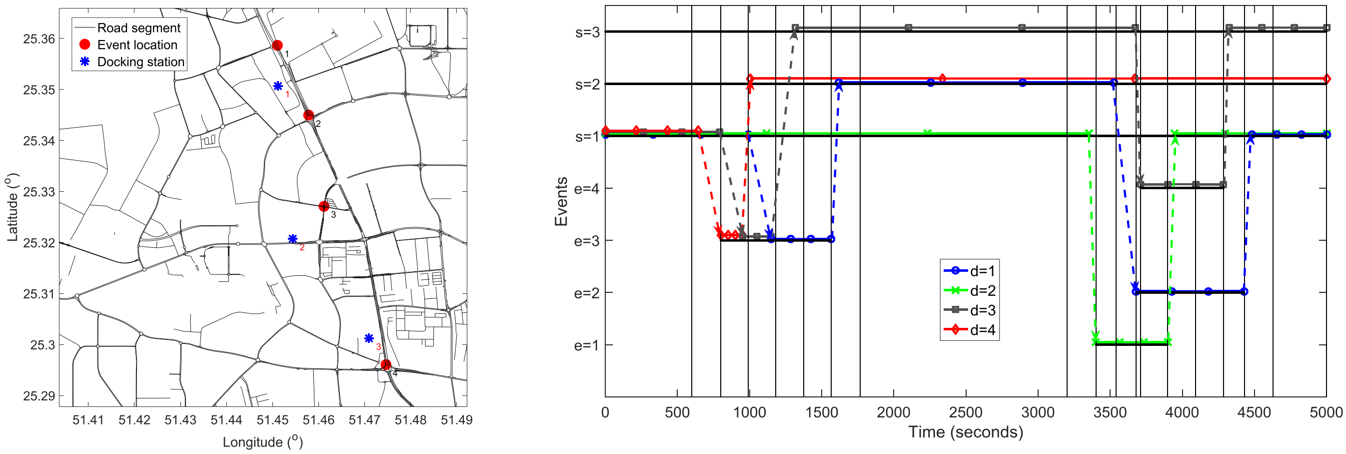
obtained with the case without RSUs as given in Fig. 2(a). However, PSO deploys the remaining docking station to another region in order increase the total coverage efficiency by avoiding redundant placement with other docking stations and RSUs.

C. RESULTS FOR THE UAV SCHEDULING

In Fig. 5, we illustrate the optimal scheduling solution applied for a scenario involving $E = 4$ events and $M = 4$ UAVs. We assume that $S = 3$ docking stations are active to replenish the UAVs batteries. Their locations are obtained using the PSO algorithm. The locations of the docking stations and the events are given in Fig. 5a while their temporal parameters are provided in Fig 5b. We denote this scenario by "Scenario 1". Event $e = 3$ starts at $\tau_3 = 800$ seconds and remains about 12 minutes while the other events $e = 1, 2, 4$ occur starting from 3400 seconds, have different durations, and overlap during some of the time slots (from $k = 12$ to $k = 15$). Notice the time horizon is divided into multiple time slots following a dynamic discretization method that takes into account the temporal and spatial characteristics of the events. More details about the dynamic discretization method can be found in [28]. The road network operator is using $D = 4$ UAVs assumed to be initially placed at docking station 1 ($s = 1$) and have completely discharged batteries. We assume that the charging power at all the docking stations are the same and are equal to $P^{ch} = 40$ W. Recall that the battery capacity is $\bar{B}_d = 27$ Wh, $\forall d$.

Table 3 is provided to explain the results provided in Fig 5. It indicates the battery level, the consumed energy, and the actions taken the UAVs UAV during each time slot. At the beginning the UAVs do not have sufficient time to reload their batteries so they cannot cover event 3 solely due to the low charging power P^{ch}. Therefore, three UAVs, namely $d = 4$, $d = 3$, and $d = 1$, are successively used. Afterwards, each of these UAVs is send to one of the docking stations depending on its future schedule. For instance, UAV 3 is sent to docking station 3 as it will cover event 4 while UAV 1 is sent to docking station 2 the closer one to event 3 and then, it will fly to cover event 2. Flying from event 3 to docking station 2 and then to event 2 is less energy consuming than flying from event 3 to docking station 1 and then event 2 due to the distances separating the locations. Finally, UAV 4 is sent back to docking station 2 the closest one and is not used in the future. UAV 2 is used to cover event 1, an event close to docking station 1. Table 3 shows that the UAVs are replenishing their batteries with the necessary energy to their planned trips. Replenishing extra energy is useless. During these periods, the UAVs are waiting at the docking stations. Finally, notice that, during the same time slot, the UAVs may have two statuses, i.e., UAV 1 is flying from docking station 1 to event 3 to cover it during time slot $k = 4$. The total energy consumption of the UAVs during the whole period of 5000 seconds is equal to 58.3 Wh.

In Fig 6, we investigate another scenario denoted by "scenario 2". We keep the same locations of the events while we change their temporal characteristics as shown in Fig. 6b. We also reduce the number of docking stations to $S = 2$. Two UAVs $D = 2$ are employed where $d = 1$ is placed in docking station 1 and $d = 2$ is placed in docking station 2. The charging power level is set to $P^{ch} = 60$ W. The time horizon is 6000 seconds. Events 3 and 2 are relatively long

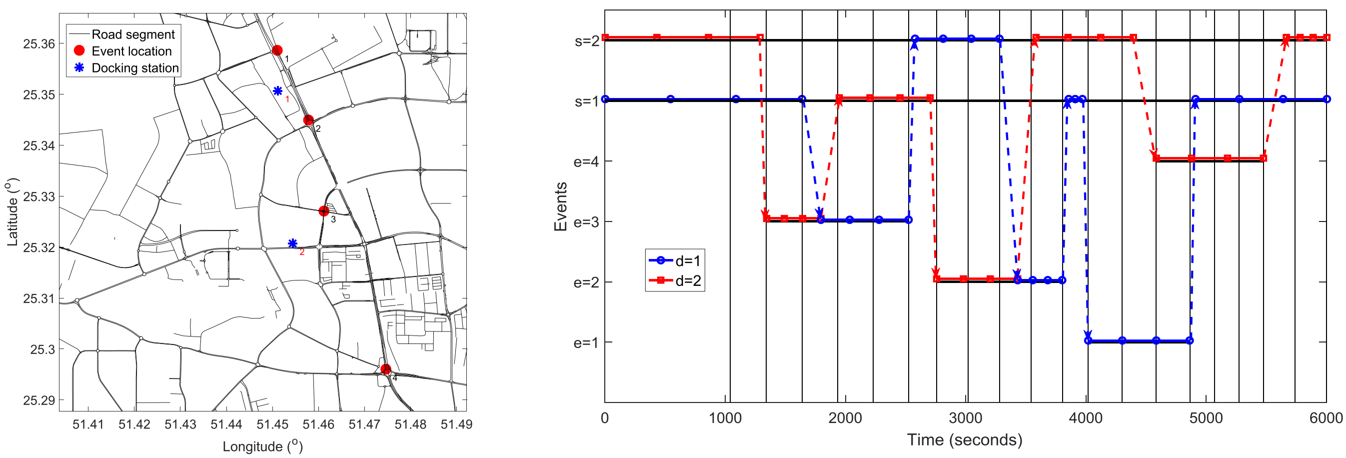


(a) (b)

FIGURE 5. Scenario 1: Optimized scheduling using $D = 4$ UAVs to cover $E = 4$ events with $S = 3$ docking stations. Event and docking station locations (left), UAV scheduling (right), dashed lines indicate the trips of the UAVs. (a) event and docking station locations, (b) UAV scheduling.

TABLE 3. Scheduling details of scenario 1 (DS: Docking Station, C: Covering, W: Waiting, F: Flying).

	Time Interval k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
UAV 1	Batt. lev. (Wh)	0	6.66	8.89	11.01	8.11	5.05	1.99	1.16	17.12	19.34	19.16	17.08	16.56	13.54	10.44	7.35	5.05	5.92
	Cons. Ener. (Wh)	0	0	0	2.9	3.06	3.06	0.82	0	0	0.18	2.08	0.52	3.01	3.1	3.1	2.29	0.78	0
	Actions	DS1	DS1	DS1	F+C3	C3	C3	F+DS2	DS2	DS2	DS2+F	F	C2	C2	C2	C2	C2	F+DS1	DS1
UAV 2	Batt. lev. (Wh)	0	6.66	8.89	11.01	13.14	13.14	15.26	15.26	15.26	14.53	12.31	10.01	9.57	6.56	5.82	5.82	7.42	9.64
	Cons. Ener.(Wh)	0	0	0	0	0	0	0	0	0.73	2.22	2.22	0.52	3.01	0.73	0	0	0	0
	Actions	DS1	DS1	DS1	DS1	DS1	DS1	DS1	DS1	DS1+F	C1	C1	C1	C1	F+DS1	DS1	DS1	DS1	DS1
UAV 3	Batt. lev. (Wh)	0	6.66	8.71	5.8	2.77	0.76	2.89	5.11	5.11	7.33	8.87	10.26	9.78	6.76	3.66	0.57	0	0
	Cons. Ener.(Wh)	0	0.10	2.91	3.02	2.01	0	0	0	0	0	0.08	0.48	3.01	3.1	3.1	0.57	0	0
	Actions	DS1	DS1+F	F+C3	C3+F	F+DS3	DS3	DS3	DS3	DS3	DS3	DS3+F	F	C4	C4	C4	F+DS3	DS3	DS3
UAV 4	Batt. lev. (Wh)	0	5.06	3.21	0.19	0	0	2.12	4.34	4.34	4.34	4.34	4.34	4.34	4.34	4.34	6.5	6.5	6.5
	Cons. Ener. (Wh)	0	2.34	3.01	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Actions	DS1	DS1+F	C3+F	F+DS2	DS2													



(a) (b)

FIGURE 6. Scenario 2: Optimized scheduling using $D = 2$ UAVs to cover $E = 4$ events with $S = 2$ docking stations. Event and docking station locations (Left), UAV scheduling (Right), dashed lines indicate the trips of the UAVs. (a) event and docking station locations, (b) UAV scheduling.

compared to the battery capacity and the charging power level of the UAVs. Hence, both UAVs are successively used to cover them. UAV 2 starts covering event 3 as it is the closest one so it can allow UAV 1 to charge its battery to the maximum as shown in Table 4 where the battery level reached 25.24 Wh during time slot $k = 4$. The UAV 2, then, moves to docking station 1 to reload its battery and starts covering event 2 during time slot $k = 8, 9, 10$ for around 675 seconds and then, returns to docking station 2 as it will serve event 4 in the future. On the other hand, after completing the coverage of event 3, UAV 1 completes serving event 2 after passing by docking station 2. Then, it covers solely event 1 which is the shortest event. Notice that, during time slot $k = 12$, UAV 1 has three actions: flying from event 2 to docking station 1,

TABLE 4. Scheduling details of scenario 2 (DS: Docking Station, C: Covering, W: Waiting, F: Flying).

	Time Interval k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
UAV 1	Batt. lev. (Wh)	0	17.38	22.38	25.24	20.67	15.95	11.22	13.36	17.72	22.08	18.04	13.86	14.24	9.74	5.23	0.73	2.56	6	6	6
	Cons. Ener. (Wh)	0	0	0	4.57	4.72	4.72	0.82	0	0	4.04	4.19	1.51	4.5	4.5	4.5	0.73	0	0	0	0
	Actions	DS1			F+C3	C3	C3	F+DS2	DS2		F+C2	C2	F+DS1+F	C1	C1	C1	F+DS1	DS1			
UAV 2	Batt. lev. (Wh)	0	17.38	20.64	15.92	11.33	12.38	17.30	19.53	15.34	11.16	7.08	10.18	13.76	18.46	17.14	12.65	9.38	6.11	2.84	1.20
	Cons. Ener.(Wh)	0	0.82	4.72	4.58	0.25	0	0.77	4.19	4.19	4.08	0.6	0	0	2.84	4.5	3.26	3.26	3.26	2.84	0
	Actions	DS2	DS2+F	C3	C3+F	F+DS1	DS1	DS1+F	C2	C2	C2+F	F+DS2	DS2		DS2+F	C4	C4	C4	C4	F+DS2	DS2

charging at docking station 1, and then, travel from docking station 1 to event 1. This shows the efficiency of the proposed approach in solving the scheduling problem in a continuous manner. In fact, unlike traditional optimization solutions where, at each time slot, a node/machine/UAV is associated to only one action/job/event, our proposed approach introduces a flexibility to the system and make the optimization close to ideal optimality, i.e., solving the problem in the continuous domain without discretization. The total energy consumption in this case is equal to 83.04 Wh which is higher than the one obtained with scenario 1. This is explained by two reasons. First, the time horizon is longer. Second, by using a lower number of UAVs, there is a need to make additional trips compared to the case where we have 4 UAVs.

VI. PERSPECTIVES AND OPEN CHALLENGES

The use of micro-UAVs has been rapidly expanding over the last few years. According to online statistics portal [46], the market for commercial UAVs is expected to reach about 13 billion US dollars by 2025 with a current total production estimated at 58.4 million US dollars. For instance, Amazon is expected to deploy around 45 thousands UAVs to reinforce its delivery fleet by 2020 under its "Amazon Prime Air" service. PwC report predicts that there are a total of about 127 billion US dollars addressable market, 60% of it is dedicated to infrastructure and agricultural applications industry alone [47]. Transportation industry is expected to be at 13 billion US dollars representing 10% of the total market. The taxi UAV is considered as one of the most prominent ITS applications helping in the reduction of congestion and traffic. The first tests of air TAXi services will be made by UBER in Los Angeles in 2020 [3].

The need to address the different challenges related to the exploitation of UAVs for ITS and smart city applications is essential to ensure the efficient operation of the flying units. In addition to the energy and scheduling issues, many other open challenges can still be confronted by UAVs. Indeed, several air fleets belonging to different owners will need to share the space to execute different tasks and missions. Therefore, it is important to guarantee a safe and collision-free navigation of the UAVs especially in urban areas. Centralized or decentralized coordination among the end-users sharing common geographical areas is required to organize the schedules of their fleet and better exploit the available charging stations.

In many applications, UAVs may have the possibility to be empowered with additional intelligence allowing them to perform in-situ decision making. Autonomous navigation might be enabled to allow UAVs determine their trajectories according to their pre-scheduled plans or to quickly react for unexpected events such as emergency situations and traffic accidents. In the latter case, the autonomous UAV will need to update its schedule and navigate towards the destination. In such situations, it is mandatory to design smart autonomous navigation algorithms guaranteeing the rapid decision making, minimizing the trip time to reach the incident location, and taking into account the different environmental factors such as obstacles and other devices, etc.

The performances of UAVs can be enhanced by increasing their missions coverage. This can be achieved by allowing the flying units to ride existing land public transport vehicles such as city buses. This will help the UAVs to increase their battery lifetime since part of the flying and hovering energy will be eliminated. Indeed, the UAVs will be carried by a city bus instead. Amazon has recently showcased the use of intermodal vehicles to support UAVs in order to accomplish the last-mile of the good delivery [48]. When they are near to reach their destination, the UAVs can be launched for last mile shipment of the goods. To enable such a concept, it is required to synchronize the operations of UAVs with those of the land transport vehicles. Hence, the trajectory of UAVs can be determined in accordance with the bus timetable. In [49], the authors has suggested such an idea in the context of video surveillance for smart cities.

In order to enable such features, adequate wireless communication infrastructure should be made available. Indeed, the level of signalling and messages to be exchanged with the ground infrastructure and among the UAVs themselves is expected to significantly grow. Different wireless communication technologies can be simultaneously deployed for this purpose such as cellular networks, Wi-Fi, and dedicated short range communications (DSRC) protocol. The choices of the used technology, the spectrum management, the data routing, and the minimization of the signaling overhead are some of the main challenges to ensure coordinated UAV scheduling and centralized, decentralized, and autonomous UAV routing and navigation.

VII. CONCLUSION

In this paper, we proposed a generic framework involving UAVs in intelligent transportation systems. Two steps for better exploitation of the UAVs are provided. The first one determines optimized emplacement of UAV docking stations in the city. Two algorithms with different conceptual constructions, i.e, PSO and PWkMeans, have been proposed to find the best locations to deploy the docking stations, that maximize the UAVs' coverage area while fulfilling the short intervention time requirement and the battery lifetime constraints.

The second step consists in a proactive scheduling approach to manage a fleet of UAVs regularly replenishing their batteries at the deployed docking stations. The scheduling framework takes into account several parameters related to the specifications of the UAVs, their batteries' level and capacity, in addition to the locations and duration of the events to be covered. An optimization problem incorporating all these features has been developed to enable parallel and sequential use of the UAVs while avoiding collisions and redundant exploitation of the resources.

Selected numerical results involving both above steps have been provided. The results showed that both algorithms that are used to calculate the optimal locations for the deployment of the docking stations achieve similar performances for different scenarios while ensuring a considerable coverage efficiency. On the other hand, we investigated the performance of the UAV scheduling problem and provided optimal energy efficient UAV-event association.

In our ongoing work, we will tackle the scenario where events occur at random locations and time instants. In such a case, a reactive scheduling approach is suitable to deal with unexpected emergency situations. The complexity related to the proposed NP-hard technique will also be mitigated by designing heuristic solutions adapted to the nature of the problem.

ACKNOWLEDGMENT

This paper was presented in part at the IEEE Vehicular Technology Conference (VTC-Spring), Sydney, Australia, June 2017 [1]. The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] H. Ghazzai, H. Menouar, and A. Kadri, "On the placement of UAV docking stations for future intelligent transportation systems," in *Proc. IEEE Vehicular Technol. Conf. (VTC-Spring)*, Sydney, NSW, Australia, Jun. 2017, pp. 1–6.

[2] *Flying New Way—A Boost for European Creativity and Innovation*, Eur. Commission Strategy Remotely Piloted Aircr. Syst., Brussel, Belgium, May 2013.

[3] J. Holden and N. Goel, "Fast-forwarding to a future of on-demand urban air transportation," Uber, San Francisco, CA, USA, Tech. Rep., Oct. 2016.

[4] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "UAV-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 22–28, Mar. 2017.

[5] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2624–2661, 4th Quart., 2016.

[6] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2016.

[7] R. Ke, Z. Li, S. Kim, J. Ash, Z. Cui, and Y. Wang, "Real-time bidirectional traffic flow parameter estimation from aerial videos," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 890–901, Apr. 2017.

[8] R. Ke, Z. Li, J. Tang, Z. Pan, and Y. Wang, "Real-time traffic flow parameter estimation from uav video based on ensemble classifier and optical flow," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 54–64, Jan. 2019.

[9] S. Minaeian, J. Liu, and Y.-J. Son, "Effective and efficient detection of moving targets from a UAV's camera," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 497–506, Feb. 2018.

[10] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: Aerial-ground cooperative vehicular networking architecture," *IEEE Veh. Technol. Mag.*, vol. 10, no. 4, pp. 36–44, Dec. 2015.

[11] H. Ghazzai, A. Feidi, H. Menouar, and M. L. Ammari, "An exploratory search strategy for data routing in flying ad hoc networks," in *Proc. IEEE Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Montreal, QC, Canada, Oct. 2017, pp. 1–7.

[12] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, "Placement optimization of UAV-mounted mobile base stations," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 604–607, Mar. 2017.

[13] H. Ghazzai, M. B. Ghorbel, A. Kadri, M. J. Hossain, and H. Menouar, "Energy-efficient management of unmanned aerial vehicles for underlay cognitive radio systems," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 4, pp. 434–443, Dec. 2017.

[14] Y. Zeng, X. Xu, and R. Zhang, "Trajectory design for completion time minimization in UAV-enabled multicasting," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2233–2246, Apr. 2018.

[15] G. Zhang, H. Yan, Y. Zeng, M. Cui, and Y. Liu, "Trajectory optimization and power allocation for multi-hop UAV relaying communications," *IEEE Access*, vol. 6, pp. 48566–48576, 2018.

[16] N. Lu, Y. Zhou, C. Shi, N. Cheng, L. Cai, and B. Li, "Planning while flying: A measurement-aided dynamic planning of drone small cells," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2693–2705, Apr. 2019.

[17] M. Ben Ghorbel, D. Rodríguez-Duarte, H. Ghazzai, M. J. Hossain, and H. Menouar, "Joint position and travel path optimization for energy efficient wireless data gathering using unmanned aerial vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2165–2175, Mar. 2019.

[18] H. Duan, P. Li, Y. Shi, X. Zhang, and C. Sun, "Interactive learning environment for bio-inspired optimization algorithms for UAV path planning," *IEEE Trans. Educ.*, vol. 58, no. 4, pp. 276–281, Nov. 2015.

[19] Q. Yang and S.-J. Yoo, "Optimal UAV path planning: Sensing data acquisition over IoT sensor networks using multi-objective bio-inspired algorithms," *IEEE Access*, vol. 6, pp. 13671–13684, 2018.

[20] J. Zhang, J. Yan, P. Zhang, and X. Kong, "Collision avoidance in fixed-wing UAV formation flight based on a consensus control algorithm," *IEEE Access*, vol. 6, pp. 43672–43682, 2018.

[21] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle routing problems for drone delivery," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 70–85, Jan. 2017.

[22] G. S. C. Avellar, G. A. S. Pereira, L. C. A. Pimenta, and P. Iscold, "Multi-UAV routing for area coverage and remote sensing with minimum time," *Sensors*, vol. 15, no. 11, pp. 27783–27803, Nov. 2015.

[23] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, Miami, FL, USA, Jun. 2017, pp. 573–580.

[24] H. Zhang, S. Wei, W. Yu, E. Blasch, G. Chen, D. Shen, and K. Pham, "Scheduling methods for unmanned aerial vehicle based delivery systems," in *Proc. IEEE/AIAA Digit. Avionics Syst. Conf. (DASC)*, London, U.K., Oct. 2014, p. 6C1.

[25] J. Kim and J. R. Morrison, "On the concerted design and scheduling of multiple resources for persistent UAV operations," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Atlanta, GA, USA, May 2013, pp. 942–951.

[26] 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," *J. Intell. Robotic Syst.*, vol. 70, nos. 1–4, pp. 347–359, Apr. 2013.

[27] B. D. Song, J. Kim, and J. R. Morrison, "Towards real time scheduling for persistent UAV service: A rolling horizon MILP approach, RHTA and the STAH heuristic," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, May 2014, pp. 506–515.

[28] H. Ghazzai, A. Kadri, M. Ben Ghorbel, H. Menouar, and Y. Massoud, "A generic spatiotemporal UAV scheduling framework for multi-event applications," *IEEE Access*, vol. 7, pp. 215–229, 2018.

[29] B. Aslam, F. Amjad, and C. C. Zou, "Optimal roadside units placement in urban areas for vehicular networks," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Cappadocia, Turkey, Jul. 2012, pp. 000423–000429.

[30] Y. Liang, H. Liu, and D. Rajan, "Optimal placement and configuration of roadside units in vehicular networks," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, Yokohama, Japan, May 2012, pp. 1–6.

[31] J. Verbeke, D. Hulens, and T. Goedemé, "How to choose the best embedded processing platform for on-board UAV image processing?" in *Proc. Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, Berlin, Germany, Mar. 2015, pp. 377–386.

[32] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Apr. 2019.

[33] G. Auer, O. Blume, V. Giannini, I. Godor, M. A. Imran, Y. Jading, E. Katranaras, M. Olsson, D. Sabella, P. Skillermark, and W. Wajda, "Energy efficiency analysis of the reference systems, areas of improvement and target breakdown," Energy Aware Radio Netw. Technol., Tech. Rep. INFSO-ICT-247733, Jan. 2012.

[34] B. S. Morse, C. H. Engh, and M. A. Goodrich, "UAV video coverage quality maps and prioritized indexing for wilderness search and rescue," in *Proc. ACM/IEEE Int. Conf. Hum.-Robot Interact. (HRI)*, Osaka, Japan, Mar. 2010, pp. 227–234.

[35] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec. 2014.

[36] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proc. 8th annual ACM-SIAM Symp. Discrete Algorithms*, New Orleans, LA, USA, Jan. 2007, pp. 1027–1035.

[37] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw*, Perth, WA, Australia, Nov/Dec. 1995, pp. 1942–1948.

[38] H. Ghazzai, E. Yaacoub, M.-S. Alouini, and A. Abu-Dayya, "Optimized smart grid energy procurement for LTE networks using evolutionary algorithms," *IEEE Trans. Veh. Technol.*, vol. 63, no. 9, pp. 4508–4519, Nov. 2014.

[39] C. A. Floudas and X. Lin, "Mixed integer linear programming in process scheduling: Modeling, algorithms, and applications," *Ann. Oper. Res.*, vol. 139, no. 1, pp. 131–162, Oct. 2005.

[40] J.-Y. Ding, S. Song, R. Zhang, R. Chiong, and C. Wu, "Parallel machine scheduling under time-of-use electricity prices: New models and optimization approaches," *IEEE. Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 1138–1154, Apr. 2016.

[41] R. Zakaria, M. Dib, L. Moalic, and A. Caminada, "Car relocation for carsharing service: Comparison of CPLEX and greedy search," in *Proc. IEEE Symp. Comput. Intell. Vehicles Transp. Syst.*, Orlando, FL, USA, Dec. 2014, pp. 51–58.

[42] (2016). *Gurobi Optimizer Reference Manual*. [Online]. Available: http://www.gurobi.com/

[43] CVX Research, Inc. (Aug. 2012). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.0*. [Online]. Available: http://cvxr.com/cvx/

[44] *OpenStreetMap*. Accessed: Nov. 25, 2018. [Online]. Available: https://www.openstreetmap.org

[45] M. Ben Brahim, W. Drira, and F. Filali, "Roadside units placement within city-scaled area in vehicular ad-hoc networks," in *Proc. Int. Conf. Connected Vehicles Expo (ICCVE)*, Vienna, Austria, Nov. 2014, pp. 1010–1016.

[46] *Commercial UAVs—Statistics & Facts*, Statista, Hamburg, Germany, 2018.

[47] M. Mazur, A. Wisniewski, and J. McMillan, "PWC global report on the commercial applications of drone technology," PricewaterhouseCoopers, London, U.K., Tech. Rep., May 2016.

[48] N. Bjone and B. C. Beckman, "Ground-based mobile maintenance facilities for unmanned aerial vehicles," U.S. Patent 15 658 080, Mar. 16, 2017.

[49] A. Trotta, F. D'Andreagiovanni, M. Di Felice, E. Natalizio, and K. R. Chowdhury, "When UAVs ride a bus: Towards energy-efficient city-scale video surveillance," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Honolulu, HI, USA, Apr. 2018, pp. 1043–1051.

HAKIM GHAZZAI (S'12–M'15) received the Diplome d'Ingenieur degree (Hons.) in telecommunication engineering and the master's degree in high-rate transmission systems from the Ecole Superieure des Communications de Tunis (SUP'COM), Tunis, Tunisia, in 2010 and 2011, respectively, and the Ph.D. degree in electrical engineering from KAUST, Saudi Arabia, in 2015. He was a Visiting Researcher with Karlstad University, Sweden, and a Research Scientist with Qatar Mobility Innovations Center (QMIC), Doha, Qatar, from 2015 to 2018. He is currently a Research Scientist with the Stevens Institute of Technology, Hoboken, NJ, USA. His general research interests include the intersection of wireless networks, UAVs, the Internet-of-Things, intelligent transportation systems, and optimization.

HAMID MENOUAR (SM'16) received the Engineering degree in computer science from the University of Sciences and Technology Houari Boumediene, Algiers, Algeria, in 2003, the DEA (M.S.) degree in systems and information technologies from the University of Technology of Compiègne, Compiègne, France, in 2004, and the Ph.D. degree in computer science from Télécom ParisTech, called at that time Ecole Nationale Supérieure des Télécommunications, Paris, France, in 2008. From 2005 to 2010, he was with Hitachi Europe, France, as a Researcher, then as a Lead of the Cooperative Systems Team. In 2010, he moved to Qatar and joined the Qatar Mobility Innovations Center (QMIC), where he is currently a Senior R&D Expert and Product Manager, leading different projects and initiatives in the areas of connected & automated vehicles, intelligent transport systems, unmanned aerial vehicles, the Internet of Things, and smart mobility.

ABDULLAH KADRI (SM'16) received the M.E.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Western Ontario (UWO), London, ON, Canada, in 2005 and 2009, respectively. From 2009 to 2012, he was a Research Scientist with the Qatar Mobility Innovations Center (QMIC), Qatar University. In 2013, he became a Senior R&D Expert and the Technology Lead at QMIC focusing on R&D activities related to intelligent sensing and monitoring using mobility sensing. His research interests include wireless communications, wireless sensor networks for harsh environment applications, indoor localization, the Internet-of-Things, green communication, applications of UAVs, and smart sensing. He was a recipient of the Best Paper Award at the WCNC Conference, in 2014.

YEHIA MASSOUD (F'15) received the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA. He has held several industrial and academic positions, including member of the Technical Staff with the Advanced Technology Group, Synopsys, Inc., Mountain View, CA, USA, a Tenured Associate Professor with the Departments of Electrical and Computer Engineering and Computer Science at Rice University, Houston, TX, USA, a W. R. Bunn Head of the Department of Electrical and Computer Engineering, University of Alabama at Birmingham, Birmingham, AL, USA, and the Head of the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA, USA. He is currently the Dean of the School of Systems and Enterprises, Stevens University of Science and Technology, Hoboken, NJ, USA. He leads research efforts in various areas of electrical and computer engineering, computing, and systems and software engineering. He has authored over 225 papers in peer-reviewed journals and conferences. He was selected as one of ten MIT Alumni's featured in the MIT EECS Newsletter, in 2012. He was a recipient of the Rising Star of Texas Medal, in 2007. He was an elected member of the IEEE Nanotechnology Council, from 2009 to 2011. He was also a recipient of the National Science Foundation CAREER Award, in 2005, the DAC Fellowship, in 2005, the Synopsys Special Recognition Engineering Award, in 2000, several best paper award nominations, and two best paper awards at the IEEE International Symposium on Quality Electronic Design, in 2007, and the IEEE International Conference on Nanotechnology, in 2011. He served as the Theme Leader of Novel Interconnects and Architectures in the SRC Southwest Academy of Nanoelectronics, from 2006 to 2011. He was named as a Distinguished Lecturer by the IEEE Circuits and Systems Society, from 2014 to 2015.