Stop & Route: Periodic Data Offloading in UAV Networks

Novella Bartolini*, Andrea Coletta†, Flavio Giorgi*,
Gaia Maselli*, Matteo Prata*, and Domenicomichele Silvestri*
* Computer Science Department Sapienza University of Rome, Rome, Italy
† J.P. Morgan AI Research, New York, USA
Email: {bartolini, giorgi, maselli, prata, silvestri.d}@di.uniroma1.it, andrea.coletta@jpmorgan.com

***Abstract*—Swarms of Unmanned Aerial Vehicles (UAVs) are a key technology to support communication in many harsh environments where fixed infrastructures (e.g., 5G) are disrupted or not available. However, the fast mobility and highly dynamic network topology pose unique challenges and require the development of novel multi-hop routing protocols. Previous work in this direction extends geographical protocols, or adapts approaches designed for Mobile Ad-hoc NETworks (MANETs), rarely taking full advantage of UAV capabilities. In this paper we introduce a novel data offloading approach, namely `Stop & Route`, that exploits the device controllable mobility to facilitate network routing. The swarm of UAVs performs data offloading synchronously and recurrently. At fixed intervals of time, the swarm interrupts the sensing mission (*stop*) and moves, as less as possible, to build a connected formation to the base station and offload the data (*route*). We provide both centralized solutions — assuming a long-range control channel — and a distributed solution — working in the absence of a control channel. By means of extensive simulations we show that our proposals outperform state-of-the-art solutions, decreasing the time taken to build a connected formation of about 25% and increasing the time spent on sensing of 14%.**

***Index Terms*—UAVs, Drones, routing, data offloading, FANETs**

I. Introduction

The past few years have witnessed an unprecedented proliferation of Unmanned Aerial Vehicles (UAVs) in people's daily lives. Thanks to their increasing capabilities and moderate cost, the use of UAVs has expanded to uncountable application scenarios, from last-mile delivery to precision agriculture, border patrolling, and many others [1]–[3].

The use of small multi-rotor UAVs in swarm formations has attracted particular attention because they are easy to assemble and quickly deployable. They are the technology of reference in monitoring critical environments where the existing communication infrastructure is damaged or knocked off. Due to their power limitations, however, small UAVs cannot rely on high data-rate long-range communications (e.g., 5G) to offload the collected data. In such scenarios, UAVs are typically required to navigate to the base station every time they need to offload monitoring data. As a consequence, they suffer fast draining of their batteries, which limits the monitoring capabilities of the network and shortens its lifetime.

One way to address these limitations is to allow the UAVs of the swarm to exploit multi-hop communication paths to deliver data to the base station. However, when deployed over large areas, the monitoring swarm may experience temporary losses of connectivity. The necessity to establish multi-hop paths for data delivery brings up the need to address the connectivity problem, and design efficient routing algorithms to deliver data packets along the swarm network. To jointly tackle both the connectivity and routing issues we propose `Stop & Route`, an innovative algorithm that periodically creates connected formations of the UAV network. Once attained a connected formation (STOP action) the UAVs can deliver their monitored data to the base station, through an efficiently selected routing path (ROUTE action).

`Stop & Route` lets the monitoring swarm move quickly upon the need, to form a connected network topology to deliver data. Thanks to its movement efficiency, it allows the network to spare communication time in favor of more device availability for monitoring and sensing tasks.

The objective of `Stop & Route` is to build the most efficient formations starting from any monitoring deployment, so as to maximize the minimum residual energy of the UAVs of the swarm, thus maximizing the network availability for the mission tasks and its lifetime.

Throughout the paper, we refer to the formation obtained during a stop phase as a *connected offloading formation*. `Stop & Route` guides the UAVs to build connected offloading formations in two modalities. The first allows for centralized coordination of the swarm. It is based on the assumption that each UAV is equipped with a long-range transmitter whose data rate, though very limited, is sufficient to establish a control channel with the base station. Through the control channel, the UAVs send and receive the necessary messages to periodically construct the desired connected formations, in a centralized manner. In centralized scenarios, the UAVs realize a connected formation based on the reception of a trigger message from the base station, hereafter called the *offloading trigger*.

The second modality, instead, considers a distributed exe-

This work is partially funded by the G5828 "SeaSec: DroNets for Maritime Border and Port Security" project under the NATO's Science for Peace Programme; † The research work was carried out when Andrea Coletta was a student at the Department of Computer Science, Sapienza University of Rome.

cution of the algorithm activities. It relaxes the assumptions on communication range and considers UAVs that are only equipped with short-range communication devices, i.e. the swarm works without any control channel. In this scenario, the network devices only rely on a local view of the achieved deployment while moving to build a connected formation. To coordinate their movements and schedule the algorithm phases, the devices adopt a loosely synchronized clock.

The `Stop & Route` algorithm provides an iterative execution of the following activities. The first is the mission activity: the UAVs execute their mission tasks, moving along the area to gather sensing data related to targets of interest. Examples of sensing applications include monitoring of large infrastructures such as roads, railways and bridges to gather data about the condition of these structures. Or post disaster: like earthquakes, hurricanes or flooding, where they can be used to monitor the damage, and help with post-disaster recovery. Upon reception of the data offloading trigger (centralized execution) or periodically (distributed execution), the UAVs execute a second activity, i.e., they build a connected offloading formation to establish communication paths with the base station. Once connected, the UAVs deliver the monitored data and resume their monitoring tasks.

The repeated execution of the two activities continues until the end of the network lifetime, that is, when the battery of the last available drone runs out of power and it must be recharged at the recharging station. The original contributions of this paper are the following:

- We define and provide a Mixed Integer Linear Programming (MILP) formulation of the Optimal Connected Offloading problem to allow the coordination of a UAV swarm for communication-constrained sensing missions.
- We provide two polynomial-time centralized solutions, Greedy Assignment Algorithm (GAA) and Tree Contraction Algorithm (TCA), to efficiently solve the Connected Offloading problem. We theoretically analyze the complexity of the two algorithms and give a performance bound on the maximum amount of distance traveled by the UAVs under the use of TCA.
- We provide a polynomial-time, distributed solution, called Distributed Gathering Algorithm (DGA), based on a loose device synchronization, and a parametric discretization of the area of interest.
- We perform extensive simulations to compare `Stop & Route` with state-of-the-art solutions. The experiments highlight that `Stop & Route` outperforms previous approaches by promptly providing connected formations that ensure lower formation times, in favor of higher availability for the mission, and longer network lifetime.

II. Related Work

The problem of exploring an area of interest with sensor networks gained a lot of attention in the literature. In order to minimize the accumulated age of the information during the monitoring activity, the problem is augmented with communication requirements. We focus on exploration pursued with multi-robots, in particular, UAVs in communication-restricted contexts. Most of the works in the literature focus on terrestrial robots, and they cannot directly be applied to UAV squads. The latter have unconstrained mobility, different energetic constraint, and requirements. For example, the amount of energy spent by UAVs in a still hovering position is not negligible as it would be for terrestrial robots. In this work, we focus on event-based, periodic connectivity, which imposes UAVs to offload their discoveries to the base station upon periodic offloading triggers. To the best of our knowledge, none of the works in the state-of-the-art addressed the same problem for UAV networks.

In [4] a taxonomy of the *communication requirements* in multi-robots exploration systems is proposed. Some exploration missions require *continuous connectivity*, that is the squad operates information gathering while remaining in partial or global communication, typically including the base station. An example of this kind is that of real-time streaming of monitoring video for human inspection at the base station. [5]–[12]. Other works require *event-based connectivity* in which UAVs regain partial or global connection with the squad, periodically or triggered by specific events [13]–[19]. Some approaches exploit temporary pairwise connectivity, intermittently present among devices [15], [19].

Dutta et al. [12] consider the problem of information collected from a polygonal environment using a multi-robot system, subject to continuous connectivity constraints. Banfi et al. [20] design exploration strategies that allow robots to coordinate with teammates to form such a network in order to satisfy recurrent connectivity constraints that is, data must be shared with the base station when making new observations at the assigned locations. Banfi et al. [21] provide a solution by modeling the signal's distribution with a Gaussian Process exploiting online different sensing strategies to coordinate and guide robots during data acquisition. Reich et al. [6] explore distributed mechanisms for maintaining the physical layer connectivity of a mobile wireless network while still permitting significant area coverage. Bartolini et al. [22] consider critical scenarios with a squad of UAVs requiring to autonomously inspect an area of interest under uncertainty of time and location of target events ensuring maximum coverage of event monitoring with minimum average inspection delay. Banfi et al [13] study effective multi-robot exploration strategies under recurrent connectivity by considering a centralized and asynchronous planning framework. They provide both the problem formalization of selecting the optimal set of locations robots should reach, the exact formulation to solve it and an approximation algorithm to obtain efficient solutions with a bounded loss of optimality. Tan et al. [5] present two heuristics based on virtual forces, with the goal to maximize sensing coverage and also guarantee continuous connectivity at the cost of a small moving distance. The heuristics do not need any knowledge of the field layout. The latter two works are used as a comparison in the experimental analysis.

III. System Model and Assumptions

We consider the scenario in which a swarm of UAVs denoted with the set $\mathcal{U} = \{u_1, \ldots, u_m\}$, is cooperatively pursuing information-gathering over an area of interest. We denote with $d_{\max}$ the longest distance between two points within the area. The control base station, denoted by σ, is placed within the area for squad coordination and acts like a sink for the information gathered from the squad. Every UAV is equipped with a GPS module and sensors to bypass obstacles in the area with ease. We denote with $p(u) \in \mathbb{R}^2$ the position of UAV u in space, analogously $p(\sigma)$ for the base station. We further denote with $b_u(t)$ the available energy of UAV u at time t, and with β_0, β_1 the energy consumption factor per unit of time spent on hovering by the UAVs and per unit of traveled distance, respectively. We denote by $h(u)$ the time spent hovering by a UAV u.

As we consider harsh environments we cannot assume the presence of any reliable long-range communication infrastructure, such as 5G. Therefore, to send sensed data to the base station, the UAVs use short-range radio transceivers (e.g., WiFi) over multi-hop routes. This communication channel, called ***primary channel***, allows communication only between neighbor nodes, ensuring a high data rate and low energy consumption. A disk model is adopted, we denote with r^{com} the transmission range of UAVs.

In some cases, we consider also the availability of a low data-rate long range communication technology, such as LoRa, that allows the transmission of control packets to the entire network. We call ***control channel*** this long-range communication technology. Both the UAVs and the base station use this channel to transmit simple commands and coordinate with each other.

To guarantee periodic information offloading at the base station, the UAVs *stop* their monitoring activity when triggered by the base station through the control channel (*offloading trigger*) and gather towards it to *route* sensing data creating a connected offloading formation.

Definition III.1 (UAVs Communication). *A pair of UAVs u_i, u_j is said to be in communication **com**($p(u_i)$, $p(u_j)$) if and only if,*

1) $\|p(u_i) - p(u_j)\| \leq r^{\text{com}}$ or
*2) $\exists\, u_k$, **com**$(p(u_i), p(u_k))$ and **com**$(p(u_k), p(u_j))$*

Definition III.2 (Connected Offloading Formation). *A connected offloading formation $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is a function mapping the coordinates of every UAV $u \in D \subseteq \mathcal{U}$ to new coordinates $f(p(u))$ such that **com**$(f(p(u)), p(\sigma))$.*

Definition III.3 (Optimal Connected Offloading Formation). *An optimal connected offloading formation $f^* : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ upon an offloading trigger at time t, is a coordinates map such that,*

$$\max \min_{u \in \mathcal{U}} \; b_u(t) - \beta_0 h(u) - \beta_1 \|p(u) - f^*(p(u))\| \quad (1)$$

it maximizes the minimum residual energy for the UAVs.

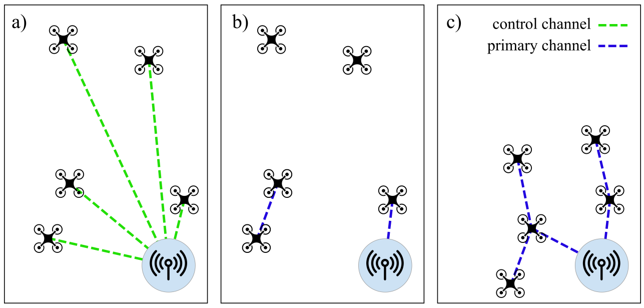

Fig. 1: Control and primary channels in a monitoring mission.

Notice that the UAVs in the swarm reach their target position $f(p(u))$ at different times, depending on their starting position $p(u)$ and traveling speed. Thus the hovering time spent by all the UAVs waiting for the last UAV to join the formation has an impact on the solution and will be taken into account in the optimal formulation presented in the next section.

We study two different ways to build connected offloading formations. First, we consider a ***centralized*** approach in which the UAVs and the base station are equipped with both the primary and the control channels. The base station and the UAVs use the control channel to build the offloading formation: the base station broadcasts the stop command, the UAVs reply by sending their positions, and the base station in turn sends the connected offloading formation. We then consider a ***distributed*** approach in which only the primary channel is available. In this case, the UAVs have only a partial view of the network (i.e., the neighborhood), and use their local information to derive a connected offloading formation in a distributed way.

Figure 1.a, shows a squad of 5 UAVs performing a monitoring mission over an area of interest. They are all connected with the base station through the control channel. Figure 1.b shows neighbor UAVs communicating over the primary channel. Figure 1.c shows a possible offloading formation, in which each UAV has an offloading route to the base station. Data offloading takes place over the primary channel.

IV. Optimal Problem Formulation

To obtain optimal connected offloading formations we propose a solution based on linear programming. We denote by $\hat{p}(u)$ the decision variable representing the target position of UAV u to generate a connected offloading formation. We first constraint the new position of the UAV to stay within the area of interest,

$$\hat{p}(u) \text{ in AOI } \forall u \in \mathcal{U} \quad (2)$$

We constraint the points that are reachable with the available energy as:

$$\|p(u) - \hat{p}(u)\| \cdot \beta_1 \leq b_u(t), \;\; \forall u \in \mathcal{U}. \quad (3)$$

Then, we add a constraint to impose the network connectivity of the new formation, to allow communication among all the UAVs and the base station. Let $\mathcal{U}^+ = \mathcal{U} \cup \{\sigma\}$. To constraint connectivity between the UAVs and the base station, we consider that each UAV generates a unit of flow and that the base station wants to receive m units. Let $\hat{s}_{ij}$ be the decision variable representing the information flow from UAV u_i to u_j. We constraint the base station to receive m units of flow as follows:

$$\sum_{i \in \mathcal{U}^+} \hat{s}_{i\sigma} = m \tag{4}$$

and to produce 0 units of flow as follows:

$$\sum_{i \in \mathcal{U}^+} \hat{s}_{\sigma i} \leq 0 \tag{5}$$

Then, we impose that UAVs generate a flow only towards their neighbors. We add an auxiliary binary variable γ_{ij} s.t. $\gamma_{ij} = 1 \iff \|\hat{p}(u_i) - \hat{p}(u_j)\| \leq r^{\text{com}}$, defined as follows,

$$\begin{aligned} \|\hat{p}(u_i) - \hat{p}(u_j)\| - M \cdot (1 - \gamma_{ij}) \leq r^{\text{com}} \\ \|\hat{p}(u_i) - \hat{p}(u_j)\| + M \cdot \gamma_{ij} \geq r^{\text{com}} \\ \gamma_{ij} \in \{0, 1\} \end{aligned} \tag{6}$$

for each i, j in $\mathcal{U}^+$ when M is an upper bound variable. Then, we impose that the flow is given by:

$$\sum_{j \in \mathcal{U}^+} \hat{s}_{ij} \leq \hat{\gamma}_{ij} \cdot m, \ \forall i \in \mathcal{U} \tag{7}$$

an edge can have a flow only if the path length between i and j is less than the communication range. Finally, to enforce connectivity we impose flow conservation through the following constraint:

$$\sum_{j \in \mathcal{U}^+} \hat{s}_{ij} = \sum_{j \in \mathcal{U}^+} \hat{s}_{ji}, \ \forall i \in \mathcal{U} \tag{8}$$

which imposes that, for each out-edge from i the flow is increased by the UAV, only if i and j otherwise. These three constraints enable the connectivity of the formation.

Let w be the variable representing the maximum travel time to reach the target point, defined as $w \geq \|p(u) - \hat{p}(u)\|/v \ \forall u \in \mathcal{U}$ with v the uniform speed of the UAVs. Thus the hovering time for a UAV is defined as

$$h(u) = w - \|p(u) - \hat{p}(u)\|/v \tag{9}$$

We finally constraint ω to be least residual energy as:

$$\omega \leq \ b_u(t) - \beta_0 h(u) - \beta_1 \|p(u) - \hat{p}(u)\|, \ \ \forall u \in \mathcal{U} \tag{10}$$

The objective function is as follows:

$$\max \omega \tag{11}$$

We approximate the euclidean distance, which is not linear to compute, using the approximation function proposed by [23].

V. Centralized Heuristics

The MILP formulation presented in the previous section is solvable in polynomial time. However, in many scenarios, the computational time for an optimal connected offloading formation is unacceptable. Figure 3 shows an improvement in computational time that goes from 3 to 4 orders of magnitude for the heuristics w.r.t. the MILP solution. This problem is exacerbated by the fact that during a mission the UAVs are called to offload data several times, thus requiring the base station to calculate a new optimal formation each time. After sending their positions to the base station the UAVs have to wait (in hovering) the time needed by the base station to calculate the optimal formation online. This amount of time not only has a catastrophic impact on the delay to build the offloading formation, but also requires substantial energy consumption due to UAVs hovering. To face this problem, we propose efficient heuristics and evaluate their computational complexity.

Algorithm 1: Greedy Assignment Algorithm (GAA)

Input: A set of UAVs $D(t) \subseteq \mathcal{U}$ under offloading trigger at time t
Output: An Connected Offloading Formation f

```
f(p(u)) ← ∅ ∀u ∈ D(t)
R ← F(Γ({σ}))
while ∃ u ∈ D(t) s.t. f(p(u)) = ∅ do
    (u*, r*) ← arg max_{r∈R, u∈D(t)} Θ(D(t) − {u}, F(R ∪ Γ({r})))
    R ← R ∪ F(Γ(r*))
    D(t) ← D(t)/{u*}
    f(p(u*)) ← r*
return f
```

A. Greedy Assignment Algorithm

We first propose the Greedy Assignment Algorithm (GAA). Algorithm 1 illustrates GAA formally. It considers the set of UAVs, $D(t) = \{u \in \mathcal{U} \mid b_u(t) > 0\}$ having a sufficient residual battery to go back to the base station for recharging. UAVs that deplete their battery, go back to the base station and are no longer available for the monitoring mission. We consider a regular hexagonal tiling of the area of interest and refer to the centers of the hexagons as rendezvous points in $\mathcal{R} \subset \mathbb{R}^2$. This set is such that taken two rendezvous points belonging to any two adjacent tiles $r_1, r_2 \in \mathcal{R}$, $\boldsymbol{com}(r_1, r_2)$. The main idea of GAA is to iteratively select for each UAV a rendezvous point that allows connection with the base station and has the least impact on the battery. First, we define $\Theta(D(t), R)$ the residual energy of the least energetic UAV of those in $D(t)$ available at time t, after reaching the rendezvous point in R maximizing the residual energy, formally:

$$\Theta(D(t), R) = \min_{u \in D} \max_{r \in R} b_u(t) - \|p(u) - p(r)\| \cdot \beta_1 \tag{12}$$

Let $X \subseteq \mathcal{R}$ be a set of rendezvous points. We denote by $\Gamma(X) \subseteq \mathcal{R}$ the set of rendezvous points adjacent to those in the set X. We define $F(X) \subseteq \mathcal{R}$ the frontier of X, to the set of rendezvous points lying on the perimeter of the convex hull induced by the rendezvous points in X. GAA iteratively builds the connected offloading formation f, by mapping at each while loop iteration a UAV u^* to a target rendezvous point r^*, chosen in the neighborhood of connected rendezvous points.

Proposition V.1. *The set of candidate rendezvous points at each iteration is upper bounded by m.*

Proof. The number of rendezvous points in R is upper bounded by the cardinality of the union of the adjacent hexagonal tiles of the UAVs positions and of the base station, that is $\mathcal{O}(6(m+1)) = \mathcal{O}(m)$.

Theorem V.2. *GAA computational complexity is $\mathcal{O}(m^5)$.*

Proof. Referring to Algorithm 1, the while loop iterates until all the UAVs are not mapped to a target rendezvous point. A map happens once at each iteration, the while loop lasts for $\mathcal{O}(m)$ iterations. The Θ function is called for every candidate rendezvous point among the $\mathcal{O}(m)$ in the growing frontier R (see Proposition V.1), and for every UAV among the $\mathcal{O}(m)$. This function has $\mathcal{O}(m^2)$ computational complexity. It follows that the overall computational complexity of GAA is polynomial in the number of UAVs and is $\mathcal{O}(m^5)$. □

B. Tree Contraction Algorithm

To lift the assumption of map discretization and improve time complexity, we propose the Tree Contraction Algorithm (TCA). Algorithm 2 illustrates TCA formally. It consists in contracting the edges of a tree, by moving the UAVs along the most convenient route toward the base station and allowing communication through the primary channel with the whole squad. The algorithm computes the Delaunay graph $G = (P, E, W)$, derived from the points $p(u)\ \forall u \in D(t) \cup \{\sigma\}$, with $W : (n_1, n_2) \in E \rightarrow \|p(n_1) - p(n_2)\|$. Then the algorithm derives from G the Euclidean Minimum Spanning Tree (EMST) $T = (P, E')$ which inherits from the Delaunay graph the property of geometric k-spanners. A geometric k-spanner is a graph in which all the path weights are upper bounded by k times the spatial distance between the path endpoints. As Theorem V.3 illustrates, this property is useful to have an upper bound on the distance traveled by the UAVs [24]. It follows an edges contraction phase, where the edges of T are made shorter to match the connection requirements through the primary channel. The shrinking process starts from the root of the tree T (representing the base station) to the leaves and proceeds iterating over the nodes of the tree sorted in a Breath First Search (BFS) fashion. For every UAV in $r \in P$, representing UAVs a new target position is computed. The new position is relative to the parent node in the tree $\text{parent}(r) \in P$. In particular the angle α between r and $\text{parent}(r)$ is used to determine the new position the UAV should reach to gain connectivity, i.e., the coordinates of the parent plus the dimension of the minimum communication range between r and its parent. Finally, $\gamma \in (0, 1]$ can be used to uniformly reduce communication range to guarantee a much stabler communication.

Theorem V.3. *The distance traveled by a UAV $u \in D(t)$ to connect to the base station σ according to a connected offloading formation produced by TCA is upper bounded by $1.998 \cdot \|p(u) - p(\sigma)\| - \gamma r^{\text{com}} \cdot \rho(u, \sigma, T)$.*

Proof. The EMST T computed by TCA guarantees that the path linking each node with the base station is the shortest. The EMST derived from the Delaunay graph is a geometric spanner. In such graphs the maximum distance between two nodes n_0, n_1 is no greater than $1.998 \cdot \|n_0 - n_1\|$ [25]. This property is maintained in the EMST. In the TCA contraction phase of the EMST, any path from u to σ having $\rho(u, \sigma, T)$ number of hops, is contracted in one of length $\gamma r^{\text{com}} \cdot \rho(u, \sigma, T)$ units of distance. Since we place UAVs belonging to the same path exactly at distance γr^{com} the one from the other, that distance can be excluded from the overall count because it will not be traveled by any UAV. □

Algorithm 2: Tree Contraction Algorithm (TCA)

Input: A set of UAVs $D(t) \subseteq \mathcal{U}$ under offloading trigger at time t
Output: An Connected Offloading Formation f

$P \leftarrow \{p(u)\ \forall u \in D(t) \cup \{\sigma\}\}$
$G(P, E, W) \leftarrow \texttt{DELAUNAY}(P)$
$T(P, E') \leftarrow \texttt{KRUSKAL}(G)$
$P \leftarrow \texttt{BFS}(T)$
$f(p(u)) \leftarrow p(u)\ \forall u \in D(t)$

for $r \in P$ **do**
  if $r \neq \sigma$ **then**
    if $\|p(r) - p(\text{parent}(r))\| > \gamma r^{\text{com}}$ **then**

$$\alpha \leftarrow \arctan\left(\frac{\Delta_Y(r, \text{parent}(r))}{\Delta_X(r, \text{parent}(r))}\right)$$

      $f(p(r)) \leftarrow p(\text{parent}(r)) + \gamma r^{\text{com}} \cdot \langle \cos(\alpha), \sin(\alpha) \rangle$
return f

Theorem V.4. *TCA computational complexity is $\mathcal{O}(m \log m)$.*

Proof. The Delaunay graph $G = (V, E, W)$ can be computed in $\mathcal{O}(m \log m)$. Kruskal's algorithm for the EMST runs in $\mathcal{O}(|E| \log |E|)$. Since the Delaunay graph is a planar graph, the upper bound on the number edges is $|E| \leq 3m-6$, as given by Euler's formula. It follows that Kruskal's algorithm runs in $\mathcal{O}(m \log m)$. The BFS on the EMST runs in linear time $\mathcal{O}(m)$. The shortening procedure requires $\mathcal{O}(m)$ iterations. Hence the overall computational complexity is $\mathcal{O}(m \log m)$. □

VI. Distributed Heuristics

We now consider a scenario in which the long-range control channel is not available and communication is possible only among neighboring UAVs by means of a short-range primary channel. Also, the base station can communicate only with neighboring UAVs. In this scenario, we propose two distributed algorithms that allow building a connected multi-hop offloading formation leveraging only partial information about the network.

Two types of control packets are exchanged to build a local network view. For neighborhood discovery, each UAV periodically sends an *hello packet*, containing, among other things, its identifier, time of creation, current position, and if it is connected with the base station or not. The base station instead notifies its presence by periodically sending heartbeat packets. When a UAV receives a heartbeat packet, it rebroadcasts it to all the neighbors to notify its connection with the base station. We assume that the UAVs are lazy-synchronized and at every Δ interval they stop their monitoring activity and gather towards the base station to route sensing data by creating a connected offloading formation.

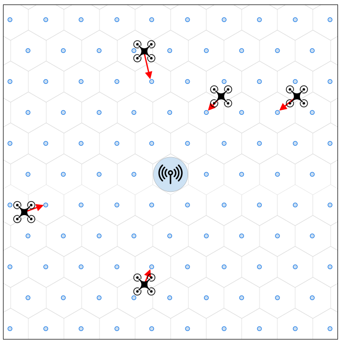

Fig. 2: ρ-*grid* rendezvous placement

A. Distributed Gathering Algorithm

Our Distributed Gathering Algorithm (DGA) is based on predefined rendezvous points and *leader-follower* strategy. We denote with $\mathcal{R}$ the set of predefined rendezvous points where the UAV traffic is conveyed to facilitate the creation of the connected formation. When a UAV u has to build a connected formation, it moves towards the rendezvous point that is closest to it and geographically closer to the base station. When u has reached the rendezvous point, three cases may happen. First, u is a neighbor of the base station or of a connected UAV. In this case, u has reached its point for data offloading; it remains there and broadcasts any hearth-beat packet it receives to notify its connection with the base station. Second, u is a neighbor of a non-connected UAV. In this case, u becomes a follower of this leader neighbor, meaning that u will follow the leader neighbor in case this starts moving. Third, u is isolated. Then u moves towards the next rendezvous point, and checks again which of the three previous cases is verified. Algorithm 3 presents the DGA algorithm. We consider a rendezvous placement called ρ-*grid* that is derived from a hexagonal tiling, in which all hexagons have uniform radius ρ (see Figure 2).

Algorithm 3: Distributed Gathering Algorithm (DGA)

Input: A UAV u running the algorithm
Output: Action for the UAV u

$\Gamma(u)\{\leftarrow u' \in \mathcal{U}\ :\ \mathbf{com}(u,u')\ \}$

if $|\Gamma(u)| > 0$ **then**
  if $\exists u' \in \Gamma(u) \mid \boldsymbol{com}(u',\sigma)$ **then**
    stop
  else
    leader $\leftarrow \arg\min_{u'\in\Gamma(u)} \|p(u') - p(\sigma)\|$
    follow leader
else
  if $\boldsymbol{com}(u,\sigma)$ **then**
    stop
  else
    $\Pi \leftarrow \{r \in \mathcal{R} \mid \|p(r) - p(\sigma)\| < \|p(u) - p(\sigma)\|$
    reach $\arg\min_{r\in\Pi} \|p(r) - p(u)\|$

VII. Performance Evaluation

We now evaluate our proposed algorithms through simulations and compare them with state-of-the-art approaches. We use a custom destined discrete-time simulator written in Python for experimenting with routing protocols and path planning for UAV networks. We run our tests on an i9-10920X 12 core (24 thread) up to 4,60 GHz and 256 GB RAM.

Scenarios. For all tests, we consider an area of interest of 1500 *m* x 1500 *m*, with the UAVs moving according to a random way-point mobility model, at a constant speed of 8 *m/s* under constant energy consumption factors β_0, β_1. The base station is located in the middle of the area and each simulation starts with UAVs placed in random positions. In our experiments we let the number of UAVs vary from 5 to 25. We consider two scenarios.

In the first scenario, we evaluate centralized heuristics. UAVs are equipped with both primary and control channels, having communication ranges 140 *m* and 6 *km* respectively. Connected offloading formations are computed according to the optimal formulation OPT (see Section IV), GAA and TCA (see Section V) and the Asynchronous Multi-robot Exploration Algorithm [13]. The latter is a topology formation algorithm for terrestrial sensor networks. It exploits Steiner trees to link a set of candidate locations to the base station using robots as relays. The new positions are computed iteratively depending upon the number of nodes to connect. For short we refer to it as AME. For the optimal solution, we used the Gurobi solver [26].

In the second scenario, UAVs are equipped only with the primary channel. Here we compare DGA (see Section VI) with Connectivity-Preserved Virtual Force (CPFV) and its floor-based scheme (FloorCPVF) [5]. The latter are two state-of-the-art algorithms for decentralized search for connected offloading formations. They consider mobile sensors moving to re-establish global connectivity with the base station with the goal of maximizing sensing range and residual energy, with total communication as a constraint. Disconnected mobile sensors move toward the base station according to a lazy movement strategy, which consists in getting closer to it and halting for a small amount of time recurrently with the hope of meeting other sensors; stopping when it finds a connected sensor. The virtual forces method is then applied to maximize area coverage. The FloorCPVF variant imposes fixed corridors for the sensors to follow, that are placed in such a way as to limit sensors' overlap.

Metrics. We measure the performance of the algorithms in terms of:

Sensing Time (seconds) defined as the average time spent by UAVs in sensing during the whole mission. It includes the time to reach targets plus the time to inspect them.

Computational Time (seconds) describes the time needed to compute the connected topology, it is computed only for the centralized solutions.

Building Time (seconds) is the time required for the UAVs to build a connected offloading formation. In centralized approaches, it is measured as the amount of time passing between when all the UAVs received their offloading coordinates, and when they are all connected. In distributed approaches, the

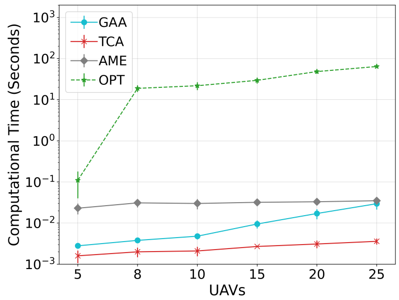

Fig. 3: Computational Time (seconds) for centralized approaches

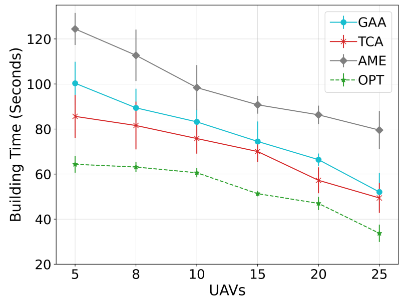

Fig. 4: Topology building time (seconds) for centralized approaches

building time is measured as the time passing between when the UAVs stop their sensing activity and when they are all connected.

A. Results for Centralized Algorithms

We recall that in centralized approaches the base station has a global view of the network. When it triggers data offloading, it calculates the connected offloading formation and sends the coordinates to the UAVs, which in the meantime have to hover on their position. The time required to calculate the offloading formation is thus a critical aspect as it determines the waiting time for the UAVs. Once received the coordinates, they can move toward the indicated point. When data offloading is completed, each UAV can return to perform sensing.

Figure 3 shows the computational time required by centralized approaches to calculating the offloading formation. Results demonstrate that the OPT solution requires long computational time even for small squads of UAVs (from $18.6s$ in the case of 8 UAVs up to $64.2s$ for 25 UAVs on average), making its *online* applicability very limited. Our GAA and TCA algorithms instead present very low computational time. TCA for instance is able to calculate an offloading formation in $0.0036s$ in the case of 25 UAVs, with a 10-fold improvement with respect to the AME algorithm.

The significant computational time reduction does not cause any performance deterioration. Figure 4 shows that both GAA and TCA are faster than AME in building the offloading formation independently of the number of UAVs: with TCA the UAVs build a formation in only $85s$ in the case of 5 UAVs, while AME takes over than $120s$ for the same number of UAVs. The building time decreases significantly when the number of UAVs increases: TCA employs less than $50s$ when the squad is composed of 25 UAVs, decreasing the building time of 38% with respect to AME. The OPT is faster than any other solution but it pays in computational time as seen before (Fig. 3).

Being faster in offloading operations (with shorter computational and building time), both GAA and TCA can devote more time to sensing with respect to AME. Figure 5 shows that GAA and TCA dedicate 14% and 10% more time respectively to sensing than AME, while OPT's performance is dropped down due to the long computational time.

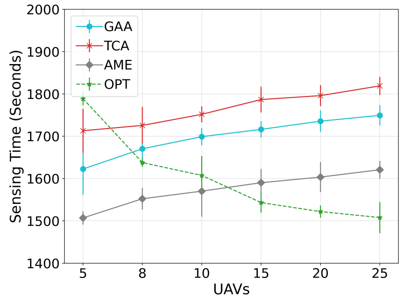

Fig. 5: Sensing Time (seconds) for centralized approaches

Thus, our first set of results clearly shows the benefits of our TCA algorithm which is faster in calculating and building a connected formation and hence it can spend more time in performing sensing.

B. Results for Distributed Algorithms

We now evaluate our distributed DGA-grid algorithm and compare it with the two state-of-the-art solutions, CPVF and Floor-CPVF [5]. Figure 6 shows the formation building time by varying the number of UAVs. CPVF and Floor-CPVF take more than $120s$ to build a formation of 5 UAVs, while this time decreases to $85-90s$ when the number of UAVs increases to 25. DGA-grid is always faster than CPVF and Floor-CPVF: it builds a connected formation of 5 UAVs in $100s$ and this time decreases to $60s$ when there are 25 UAVs in the squad, achieving a maximum improvement of 25% in case of 8 UAVs.

The time gained while building the connected formation can be used to perform sensing. Figure 7 shows that DGA-grid performs a better utilization of mission time as it increases the sensing time of 7% with respect to CPVF.

VIII. Conclusions

We presented `Stop & Route`, a novel approach to handling a squad of UAVs under recurrent connectivity constraints. We assume two types of network UAV equipment: long-range and short-range communication hardware. In the first scenario, UAVs are equipped with both hardware components. In this case, we propose two different centralized approaches, GAA and TCA, resorting to a central unit to compute the connected offloading formation starting from the

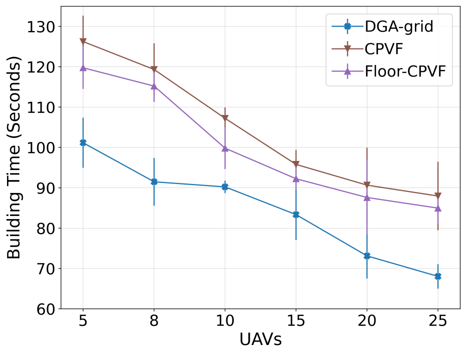

Fig. 6: Building Time (seconds) for distributed approaches

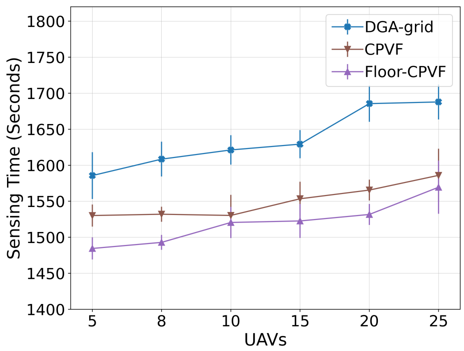

Fig. 7: Sensing Time (seconds) for distributed approaches

most recent UAV positions. The other scenario instead involves only short-range hardware components. In this scenario, we provide a decentralized solution, called DGA, to build the connected offloading formation. Results show that our proposed algorithms, GAA and TCA, outperform the state-of-the-art centralized algorithm (AME) reaching up to 29% and a 33% improvement in building time. We also show that both our algorithms bring an improvement in sensing time over the state-of-the-art of about 14% for TCA and 10% for GAA, reducing the computational time up to one order of magnitude. Regarding decentralized approaches, we tested the DGA-grid rendezvous placement. Results show that DGA-grid improves the performance of the state-of-the-art by about 15% in building time and 7% in sensing time.

References

[1] F. Afghah, A. Razi, J. Chakareski, and J. Ashdown, "Wildfire monitoring in remote areas using autonomous unmanned aerial vehicles," *IEEE MiSARN*, 2019.

[2] M. Kulbacki, J. Segen, W. Knieć, R. Klempous, K. Kluwak, J. Nikodem, J. Kulbacka, and A. Serester, "Survey of drones for agriculture automation from planting to harvest," in *2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES)*. IEEE, 2018, pp. 000 353–000 358.

[3] N. Bartolini, A. Coletta, G. Maselli, and M. Piva, "Druber: A trustable decentralized drone-based delivery system," in *Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2020, pp. 1–6.

[4] F. Amigoni, J. Banfi, and N. Basilico, "Multirobot exploration of communication-restricted environments: A survey," *IEEE Intelligent Systems*, vol. 32, no. 6, pp. 48–57, 2017.

[5] G. Tan, S. A. Jarvis, and A.-M. Kermarrec, "Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 6, pp. 836–848, 2009.

[6] J. Reich, V. Misra, D. Rubenstein, and G. Zussman, "Connectivity maintenance in mobile wireless networks via constrained mobility," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 935–950, 2012.

[7] N. Bartolini, A. Coletta, M. Prata, and C. Serino, "On connected deployment of delay-critical fanets," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 9720–9727.

[8] J. Scherer and B. Rinner, "Persistent multi-uav surveillance with energy and communication constraints," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*. IEEE, 2016, pp. 1225–1230.

[9] ——, "Multi-robot persistent surveillance with connectivity constraints," *IEEE Access*, vol. 8, pp. 15 093–15 109, 2020.

[10] X. Liu, T. Xi, E. Ngai, and W. Wang, "Path planning for aerial sensor networks with connectivity constraints," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.

[11] Y. Marchukov and L. Montano, "Fast and scalable multi-robot deployment planning under connectivity constraints," in *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2019, pp. 1–7.

[12] A. Dutta, A. Ghosh, and O. P. Kreidl, "Multi-robot informative path planning with continuous connectivity constraints," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3245–3251.

[13] J. Banfi, A. Q. Li, N. Basilico, I. Rekleitis, and F. Amigoni, "Asynchronous multirobot exploration under recurrent connectivity constraints," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 5491–5498.

[14] G. A. Hollinger and S. Singh, "Multirobot coordination with periodic connectivity: Theory and experiments," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 967–973, 2012.

[15] A. Howard, M. J. Matarić, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, no. 2, pp. 113–126, 2002.

[16] S. Hayat, E. Yanmaz, C. Bettstetter, and T. X. Brown, "Multi-objective drone path planning for search and rescue with quality-of-service requirements," *Autonomous Robots*, vol. 44, no. 7, pp. 1183–1198, 2020.

[17] G. Danoy, M. R. Brust, and P. Bouvry, "Connectivity stability in autonomous multi-level uav swarms for wide area monitoring," in *Proceedings of the 5th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*, 2015, pp. 1–8.

[18] N. Goddemeier, K. Daniel, and C. Wietfeld, "Role-based connectivity management with realistic air-to-ground channels for cooperative uavs," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 951–963, 2012.

[19] N. Bartolini, A. Coletta, A. Gennaro, G. Maselli, and M. Prata, "Mad for fanets: Movement assisted delivery for flying ad-hoc networks," in *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2021, pp. 315–325.

[20] J. Banfi, A. Q. Li, I. Rekleitis, F. Amigoni, and N. Basilico, "Strategies for coordinated multirobot exploration with recurrent connectivity constraints," *Autonomous Robots*, vol. 42, no. 4, pp. 875–894, 2018.

[21] J. Banfi, A. Q. Li, N. Basilico, I. Rekleitis, and F. Amigoni, "Multirobot online construction of communication maps," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2577–2583.

[22] N. Bartolini, A. Coletta, and G. Maselli, "Side: Self driving drones embrace uncertainty," *IEEE Transactions on Mobile Computing*, 2021.

[23] Y. Xie, S. Zhou, Y. Xiao, S. Kulturel-Konak, and A. Konak, "A β-accurate linearization method of euclidean distance for the facility layout problem with heterogeneous distance metrics," *European Journal of Operational Research*, vol. 265, no. 1, pp. 26–38, 2018.

[24] d. B. Mark, C. Otfried, v. K. Marc, and O. Mark, *Computational geometry algorithms and applications*. Spinger, 2008.

[25] G. Xia, "The stretch factor of the delaunay triangulation is less than 1.998," *SIAM Journal on Computing*, vol. 42, no. 4, pp. 1620–1659, 2013. [Online]. Available: https://doi.org/10.1137/110832458

[26] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual," 2022. [Online]. Available: https://www.gurobi.com