FlyKites: Human-centric Interactive Exploration and Assistance under Limited Communication

Yuyang Zhang*, Zhuoli Tian*, Jinsheng Wei and Meng Guo

***Abstract*— Fleets of autonomous robots have been deployed for exploration of unknown scenes for features of interest, e.g., subterranean exploration, reconnaissance, search and rescue missions. During exploration, the robots may encounter unidentified targets, blocked passages, interactive objects, temporary failure, or other unexpected events, all of which require consistent human assistance with reliable communication for a time period. This however can be particularly challenging if the communication among the robots is severely restricted to only close-range exchange via ad-hoc networks, especially in extreme environments like caves and underground tunnels. This paper presents a novel human-centric interactive exploration and assistance framework called FlyKites, for multi-robot systems under limited communication. It consists of three interleaved components: (I) the distributed exploration and intermittent communication (called the "spread mode"), where the robots collaboratively explore the environment and exchange local data among the fleet and with the operator; (II) the simultaneous optimization of the relay topology, the operator path, and the assignment of robots to relay roles (called the "relay mode"), such that all requested assistance can be provided with minimum delay; (III) the human-in-the-loop online execution, where the robots switch between different roles and interact with the operator adaptively. Extensive human-in-the-loop simulations and hardware experiments are performed over numerous challenging scenes.**

I. INTRODUCTION

Exploration of unknown and hazardous scenes before allowing humans inside is particularly suitable for robots. For instances, fleets of UAVs and UGVs have been deployed to explore planetary caves in [12], [21]; search and rescue after earthquakes in [6]. Many collaborative exploration strategies have been proposed along with the advances in autonomous navigation and perception, e.g., [5], [11], [29], [18], [32]. They often assume an all-to-all communication among the robots, i.e., instant map sharing among the robots and with a static base station. However, this could be impractical in many aforementioned scenes where the communication facilities are unavailable or severely degraded. Namely, the robots can only exchange information via ad-hoc networks in close proximity. This imposes great challenges on the fleet coordination as communication and exploration are now closely dependent thus need to be jointly planned.

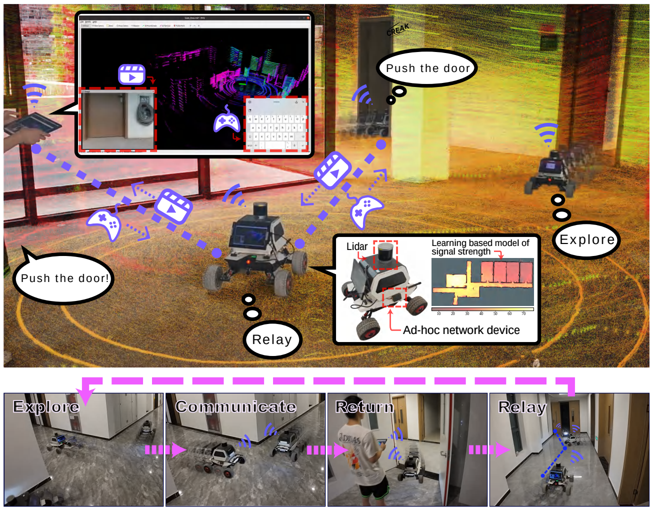

Fig. 1: Top: the FlyKites system deployed for hardware experiments. One robot explores the workspace, while two robots in relay mode to transmit: (I) live video stream from the end robot to the operator, and (II) tele-operation from the operator to the end robot to open a door. Their communication is fully distributed via ad-hoc networks, of which the quality is predicted via a learned model. **Bottom**: Three UGVs transit between exploration, intermittent communication, return to the operator, and communication relay.

Moreover, the robots often encounter situations where they need human assistance during the exploration process [17]. For instance, the planned passage might be too tight for autonomous navigation, which requires the operator to drive the robot manually [22]; the robot might encounter unidentified targets that need human inspection [3]; some objects might be interactive such as movable obstacles and doors to push open, as shown in Fig. 1; or the robots are simply stuck and need human intervention. In these cases, a reliable communication between the requested robot and the operator is crucial for a period of time until the request is resolved [13]. This however is particularly challenging to fulfill, if the robots and the operator can only communicate in close range in these extreme environments.

A. Related Work

Multi-robot exploration has been extensively studied in the literature [2], [5], [11], [18], [30], [32]. These work often assume that all robots can exchange information via wireless communication perfectly and instantly at all time. This is often impractical or infeasible without infrastructures for communication already installed. As discussed in [7], due to obstructions of obstacles in subterranean or indoor structures, the inter-robot communication is severely limited in range and bandwidth. To overcome this challenge, many recent work can be found that combines on the planning for

---

The authors are with the College of Engineering, Peking University, Beijing 100871, China. This work was supported by the National Natural Science Foundation of China (NSFC) under grants 62203017, U2241214, T2121002. *Equal Contribution. Contact: `meng.guo@pku.edu.cn`.

of inter-robot communication and autonomous exploration. The survey in [1] provides a quantitative comparison of different communication-constrained exploration strategies, including [20], [24]. Different intermittent communication strategies have been proposed w.r.t. different performance metrics [4], [8], [9], [14], [26], [28]. On the other hand, all-time fully-connected networks are imposed in [24], [31] by collaborative motion planning, and further in [15], [20] by placing front and relay nodes. However, most these works focuse one specific mode of communication, while neglecting the online interaction with human operators.

Indeed, the human operator plays an indispensable role for the operation of robotic fleets, despite of their autonomous capability. In many aforementioned scenarios, not only the operator should be aware of the system status but also directly assist certain procedures whenever necessary. Almost all aforementioned work neglects this aspect and assume a static base station for visualization, see [8], [12], [14], [19], [25], [28], yielding a rather uniform behavior of the fleet. An augmented-reality (AR) device is proposed in [23] to facilitate the human-fleet real-time interaction, which however relies on high-bandwidth communication. The aforementioned bilateral and online interactions remains largely unexplored in the literature, i.e., how different communication modes can enable these online interactions.

B. Our Method

To tackle these challenges, this work proposes a novel framework FlyKites, for the online interaction and assistance between the operator and the robotic fleet in unknown and communication-constrained environments. In particular, a generic collaborative exploration scheme is proposed for multi-robot systems with intermittent communication (called the "spread mode"), where the inter-robot communication and frontier-based exploration are jointly planned between pairs of robots. Then, to fulfill the robot requests for human assistance during exploration, a hybrid optimization problem is formulated to simultaneously optimize the relay topology, the operator path and the associated assignment of robots to relay roles, such that a reliable chain of communication can be established between the operator and the requested robot. Thus, the requested assistance can be provided with minimum delay, thus called the "relay mode". Finally, an online strategy for human-in-the-loop execution is proposed, where the robots switch between the exploration, communication and relay modes, according to the exploration progress and the online interactions. Extensive human-in-the-loop simulations and hardware experiments are performed over numerous challenging scenes.

Main contributions of this work are two-fold: (I) a novel and generic framework for the online interaction and assistance between a dynamic operator and a robotic fleet in unknown and communication-constrained environments; (II) two different modes of simultaneous exploration and communication, i.e., the spread mode and the relay mode, and the transition strategy between them.

II. PROBLEM DESCRIPTION

A. Robots and Operator in Workspace

Consider a 2D workspace $\mathcal{A} \subset \mathbb{R}^2$, of which its map including the boundary, freespace and obstacles are all unknown. A team of robots denoted by $\mathcal{N} \triangleq \{1, \cdots, N\}$ is deployed by an operator to explore the workspace. Each robot $i \in \mathcal{N}$ is capable of simultaneous localization and mapping (SLAM) with collision avoidance. Denote by $p_i(t) \in \mathcal{A}$ the 2D pose and $M_i(t) \subseteq \mathcal{A}$ the local map of robot i at time $t > 0$. Similarly, the operator has a 2D pose $p_{\mathtt{h}}(t)$ and a local map $M_{\mathtt{h}}(t)$. For brevity, denote by $\mathcal{N}^+ \triangleq \mathcal{N} \cup \{\mathtt{h}\}$. Moreover, the robots and the operator are equipped with a communication module to exchange data locally as follows.

Definition 1 (Neighbors)**.** Each robot $i \in \mathcal{N}^+$ (or the operator) can communicate with another neighboring robot $j \in \mathcal{N}^+$ (or the operator), if the communication quality between them is above a threshold, i.e., $\mathtt{Com}_{ij}(p_i, p_j, M_i) > \underline{c}$. $\underline{c}$ represents the minimum signal strength required for successful communication in a specific environment, which can be determined through experimental measurements. ■

The neighbors of robot $i \in \mathcal{N}$ at time $t > 0$ is denoted by $\mathcal{N}_i(t) \subseteq \mathcal{N}^+$, which is symmetric and time-varying. Thus, the behavior of each robot $i \in \mathcal{N}$ is determined by its timed sequence of navigation and communication events, i.e.,

$$\Gamma_i \triangleq c_i^0 \, \mathbf{p}_i^0 \, c_i^1 \, \mathbf{p}_i^1 \, c_i^2 \, \cdots, \tag{1}$$

where $c_i^m \triangleq (j,\, p_{ij},\, [t_m,\, (t_m + T_{ij})])$ is the communication event with robot $j \in \mathcal{N}_i(t)$ at location $p_{ij} \in \mathcal{A}$ during the time interval $[t_m,\, (t_m + T_{ij})]$; the navigation path $\mathbf{p}_i^m \subset \mathcal{A}$ contains the waypoints between these communication events. Similarly, the behavior of the operator is denoted by $\Gamma_{\mathtt{h}}$, which however is not fully *controllable*.

B. Online Assistance via Human-robot Interaction

During exploration, the robots may encounter situations that require human assistance, e.g., to identify a target, to take over control inputs, to manipulate an interactive object, or to resolve a temporary failure. Thus, a set of *unknown* assistance tasks for the fleet is defined as follows:

$$\Phi \triangleq \{\phi_1,\, \phi_2,\, \cdots, \phi_K\}, \tag{2}$$

where $\phi_k \triangleq (p_k,\, i_k,\, \rho_k,\, T_k)$ represents the k-th assistance task, which includes the location $p_k \in \mathcal{A}$, the robot $i_k \in \mathcal{N}$ requiring assistance, the priority $\rho_k > 0$, and the minimum required duration $T_k > 0$ for the task, $\forall k \in \mathcal{K} \triangleq \{1, \cdots, K\}$. Note that T_k is unknown and specified online by the operator.

Definition 2 (Condition for Assistance)**.** An assistance task $\phi_k \in \Phi$ is *accomplished* at time $t_k > T_k$, under the following conditions: (I) robot i_k is at the location p_k, with $p_{i_k}(t_k) = p_k$; (II) there exists a chain of neighboring robots connecting robot i_k to the operator during the time period $[(t_k - T_k),\, t_k]$, i.e.,

$$\boldsymbol{\xi}_k \triangleq i_k \, i_k^1 \, i_k^2 \, \cdots \, i_k^{L_k} \, \mathtt{h}, \tag{3}$$

where $i_k^{\ell} \in \mathcal{N}$ is the relay robot, $\forall \ell \in \widehat{L}_k \triangleq \{1, \cdots, L_k\}$ with L_k relay robots in total; and $i_k^{\ell+1} \in \mathcal{N}_{i_k^{\ell}}(t)$, $i_k^1 \in \mathcal{N}_{i_k}(t)$, $\texttt{h} \in \mathcal{N}_{i_k^{L_k}}(t)$, $\forall \ell \in \widehat{L}_k$ and $\forall t \in [(t_k - T_k),\, t_k]$. ■

In other words, the conditions for assistance require that a consistent and reliable communication chain is established between the robot that needs assistance and the operator, which is often assumed in related work [5], [11], [18], [32]. For brevity, denote by $\widehat{\Gamma}(t) \triangleq \big(\{\Gamma_i(t)\}, \Gamma_{\texttt{h}}(t)\big)$ the joint behaviors of the robots and the operator by time $t > 0$. Then, $\widehat{\Gamma}(t) \models \phi_k$ if the joint behaviors satisfy the above conditions for the assistance task $\phi_k \in \Phi$.

C. Problem Statement

The overall problem is formalized as a constrained optimization over the collaborative exploration and communication strategy over the fleet, i.e.,

$$\begin{aligned} &\min_{\{\widehat{\Gamma},\, \overline{T}\}} \; \overline{T} \\ \textbf{s.t.} \quad & \mathcal{A} \subseteq M_{\texttt{h}}(\overline{T}); \qquad &(4a) \\ & \widehat{\Gamma}(\overline{T}) \models \phi_k, \; \forall \phi_k \in \Phi; \qquad &(4b) \end{aligned}$$

where $\overline{T} > 0$ is the total time when the complete map $\mathcal{A}$ is known to the operator by (4a), and all online requests for assistance tasks are fulfilled by (4b).

III. PROPOSED SOLUTION

The solution contains three main components: (I) the distributed exploration and intermittent communication as the fundamental building block of "spread mode"; (II) the simultaneous optimization of the relay topology, the operator path, and the assignment of robots to relay roles, to fulfill the requested assistance, as the "relay mode"; and (III) the human-in-the-loop online execution, where the robots switch between different modes and interact with the operator.

A. Distributed Exploration and Intermittent Communication

1) Frontier-based Exploration: The frontier-based method in [29] allows a robot to explore the environment by repetitively reaching the frontiers on the boundaries between the explored and unexplored areas of its local map. Namely, given the local map $M_i(t)$ of robot $i \in \mathcal{N}$ at time $t > 0$, these boundaries can be identified via a Breadth-First-Search (BFS), which are then clustered to a few frontiers by various metrics [10]. Denote by $\mathcal{F}_i(t) \triangleq \{f_k\}$ the set of frontiers, where each frontier $f_k \in \partial M_i$ is on the boundary ∂M_i. Thus, $\mathcal{F}_i(t)$ is empty if $M_i(t)$ is fully explored.

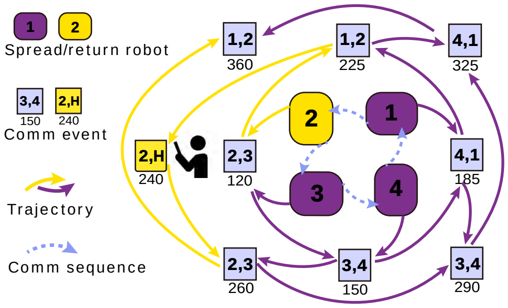

Fig. 2: The intermittent communication protocol with the ring topology: pairwise communication events (blue) and return events to the operator (yellow).

2) Intermittent Communication with Ring Topology: To ensure that the information can be propagated among the team and to the operator, the robots are required to meet and communicate via intermittent communication during exploration. As shown in Fig. 2, initially all robots are in close proximity but follows a fixed *ring* topology. Namely, each robot i only communicates with its predecessor $i-1$ and successor $i+1$, $\forall i = 2, \cdots, (N-1)$; robot N with robots 1 and $N{-}1$; and robot 1 with robots N and 2. Without the need for assistance, when robots i and j communicate at a planned event $c_{ij} \triangleq (t_{ij}, p_{ij})$ with time $t_{ij} > 0$ and location $p_{ij} \in M_i \cap M_j$, they follow these steps: (I) their local maps M_i and M_j are merged as the new local map, i.e., $M_{ij} \triangleq \texttt{merge}(M_i, M_j)$; (II) the frontiers $\mathcal{F}_{ij}$ within M_{ij} is computed; (III) the "time and place" for their *next* communication event is determined by the constrained optimization problem:

$$\begin{aligned} (\boldsymbol{\tau}_i^+,\, \boldsymbol{\tau}_j^+) &= \underset{(p_{ij}^+,\, t_{ij}^+)}{\textbf{argmax}} \Big\{ \texttt{C2VRP}\big((\boldsymbol{\tau}_i, \boldsymbol{\tau}_j), p_{ij}^+, \mathcal{F}_{ij}\big) \Big\} \\ \textbf{s.t.} \quad & p_{ij}^+ \in M_{ij}, \quad t_{ij}^+ \leqslant t_{ij} + \widehat{T}_{\texttt{h}}(t_{ij}); \end{aligned} \tag{5}$$

where (p_{ij}^+, t_{ij}^+) is the next communication event to be optimized; $\boldsymbol{\tau}_i$ and $\boldsymbol{\tau}_j$ are the current local plans of robots i and j, including the their confirmed meeting events with other robots; function $\texttt{C2VRP}(\cdot)$ stands for the constrained two-vehicle routing problem to find the updated plans $\boldsymbol{\tau}_i^+$ and $\boldsymbol{\tau}_j^+$, such that the robots can reach the maximum number of frontiers in $\mathcal{F}_{ij}$, while respecting the confirmed meeting events; and p_{ij}^+ should be chosen within the merged map, while the time t_{ij}^+ should be within a specified window $\widehat{T}_{\texttt{h}}(t_{ij})$, which is updated online as described in the sequel. In other words, the robots should collaboratively determine the next communication event, such that the exploration efficiency can be maximized. The above problem can be solved either optimally via a mixed-integer programming (MIP) solver, or approximately by solving two traveling salesman problem with time windows (TSPTW) in sequence; (IV) after obtaining the updated plans $(\boldsymbol{\tau}_i^+, \boldsymbol{\tau}_j^+)$, they depart and do not communicate until the confirmed communication event.

Remark 1. During exploration, it often occurs that robot i meets with another robot k on its way to the meeting event with robot j, which is called a *spontaneous* meeting event. In this case, they exchange their local data D_i and D_k without coordinating the next meeting event, such that the communication topology $\mathcal{G}$ can be maintained, instead of growing into a full graph. ■

3) Latency-bounded Return to Human Operator: As previously discussed, it is imperative that the robots frequently return to the operator to provide timely updates on exploration progress, system status, and any potential assistance requests. The robot designated for this task is referred to as

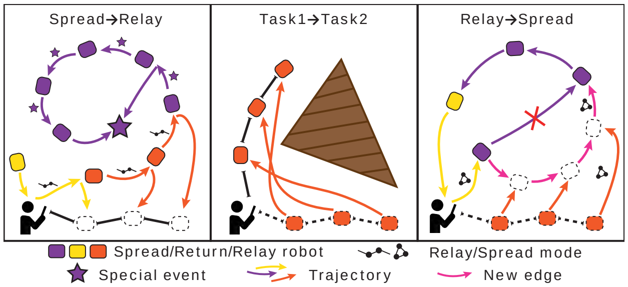

Fig. 3: Mode transitions from the spread mode to the relay mode (**left**), directly between two relay modes (**middle**), or back to the spread mode (**right**).

the *messenger*. To this end, a latency T_{h} is specified by the operator, meaning that at least one robot should return to the operator every T_{h}. This can be enforced by maintaining an estimated time stamp:

$$\widehat{T}_{\mathrm{h}}(t) \triangleq T_{\mathrm{h}} - t + \max_{i\in\mathcal{N}}\{t^i_{\mathrm{h}}\}, \qquad (6)$$

where t^i_{h} is the last time that robot i returned to the operator; and $\widehat{T}_{\mathrm{h}}(t)$ is the estimated time left for the next return. Thus, during the intermittent communication by (5), the time window $\widehat{T}_{\mathrm{h}}(t)$ is updated after exchanging $\{t^i_{\mathrm{h}}\}$. Then, each robot checks if it can return immediately to the operator and meet the latency requirement, *after* they meet at the optimized event (p^+_{ij}, t^+_{ij}). If not, it means that one of them should return to the operator before their next communication event. Since the preceding robot $i < j$ has the latest data, it would return to the operator after its next confirmed meeting event. This results in another confirmed meeting event at p_{h} and estimated time of arrival t^i_{h}, which is appended to the local plan $\boldsymbol{\tau}_i$. Consequently, the same optimization problem (5) is adopted given this updated plan to determine the next communication event for robot i and j. Similar analyses can be found in our previous work [27].

B. Hybrid Optimization for Assistance Tasks

During online exploration, the robots may encounter scenarios where they need human assistance, as defined in (2). For each assistance task $\phi_k \in \Phi$, the critical condition for its accomplishment from Def. 2 is that there exists a communication chain $\boldsymbol{\xi}_k$ in (3) from robot i_k to the operator $\mathtt{h}$, which is optimized in this part.

1) Relay Topology Optimization: To begin with, the relay topology of the chain communication is optimized. namely the positions of the relay robots. The collision-free shortest path from p_k to $\mathtt{h}$ in M_{ij} from robot i_k to the operator $\mathtt{h}$ is computed as $\mathbf{p}^\star$. Then, starting from the first anchor $p^\ell_k = \mathbf{p}^\star[k_\ell]$ where $\ell = 0$, the next anchor $p^{\ell+1}_k = \mathbf{p}^\star[k_{\ell+1}]$ is given by the largest index $k_{\ell+1} > k_\ell$ such that $\mathtt{Com}(p^\ell_k, \mathbf{p}^\star[k_{\ell+1}], M_{ij}) > \underline{c}$ holds. The above procedure terminates when $k_\ell = |\mathbf{p}^\star|$, namely the last anchor is the operator $p^{L_k+1}_k = \mathtt{h}$. The resulting topology is denoted by $\mathbf{p}_k$ for the k-th assistance task.

2) Assignment of Relay Roles: Consider that $L_k \leqslant N - 1$, i.e., there are enough robots to form the relay chain. To ensure that an assistance task can be accomplished as soon as possible, the succeeding L_k robots of the messenger in the ring topology are assigned as relay robots, for they are the earliest to finish their last meeting events. Denote by $\mathcal{N}_k \subseteq \mathcal{N}$ the set of assigned robots. The problem is to determine the optimal assignment $\Pi_k : \mathbf{p}_k \to \mathcal{N}_k$, such that the time when all assigned robots reach the anchor points is minimized, i.e.,

Algorithm 1: Online Execution and Adaptation

```
Input  : $\Phi$.
Output: $\{\Gamma_i\}$, $M_{\mathrm{h}}$.
while $M_h \neq \mathcal{A}$ do
    /* Spread Mode                                   */
    for neighbors $(i, j)$ do
        Compute $M_{ij}$ and $\mathcal{F}_{ij}$;
        Update $\Gamma^+_i, \Gamma^+_j$ by (5);
        if return event required by (6) then
            Return to operator and update $M_{\mathrm{h}}$, $\Phi$;
    /* Relay Mode                                    */
    if assistance $\phi_k \in \Phi$ known at the operator then
        Compute assignment $\Pi_k, \mathcal{I}, \mathcal{J}$ by (7);
        Robots $\mathcal{I}$ transit to the relay mode;
        if $\phi_k$ is completed then
            All robots transit back to the spread mode;
```

$$\min_{\Pi_k}\Big\{\max_{\ell\in\widehat{L}_k}\ \{t_{i^\ell_k} + T_{\mathtt{Nav}}\big(p_{i^\ell_k}(t^\ell_k),\, p^\ell_k\big)\}\Big\}, \qquad (7)$$

where $\Pi_k(p^\ell_k) \triangleq (i^\ell_k, t^\ell_k)$, $\forall \ell = 1, \cdots, L_k$; $i^\ell_k \in \mathcal{N}_k$ is the robot assigned to the anchor point p^ℓ_k; t^ℓ_k is the time when robot i^ℓ_k navigates from its final meeting event $p_{i^\ell_k}(t^\ell_k)$ towards p^ℓ_k. The above problem resembles the linear bottleneck assignment problem (LBAP), where the objective is to minimize the maximum cost of an agent-task assignment (i.e., not the summation). It can be solved in a centralized manner by the messenger, since it has obtained the planned meeting events of all other robots due to the ring topology.

3) Operator's movement: The operator may relocate upon each messenger's return to: (I) reduce relay robots when $L_k > N - 1$, by moving to an anchor point along topology $\mathbf{p}_k$; (II) shorten the messenger's return by moving in the main exploration direction, speeding exploration. The messenger records and distributes the operator's location.

C. Mode Transitions

Given the assignments, some relay robots must transition between spread and relay modes. As shown in Fig. 3, this section details the bilateral transition process.

1) Transition From Spread Mode to Relay Mode: Once the relay topology and role assignments are set, the fleet transitions to relay mode, following the same intermittent communication protocol. The key challenge is ensuring the remaining robots maintain the ring topology and continue exploration in spread mode. Assume that robot $i^\star \in \mathcal{N}$ returns to the operator; $\mathcal{I} \triangleq i^\star i^1_k i^2_k \cdots i^{L_k}_k$ are the relay robots; and $\mathcal{J} \triangleq j^1_k j^2_k \cdots j^{\overline{L}_k}_k$ are the remaining robots in the spread mode, where $\overline{L}_k \triangleq N - L_k - 1$. Then, the transition phase consists of three stages: (I) Each relay robot $i_k^\ell \in \mathcal{I}$ propagates the relay decisions to its succeeding robot $i_k^{\ell+1}$, and then directly navigates to its assigned anchor point; (II) Robot j_k^1 changes its preceding robot from $i_k^{L_k}$ to $j_k^{\overline{L}_k}$, and updates their next meeting event $c(j_k^1, j_k^{\overline{L}_k})$ by (5); (III) This updated event is propagated among $\mathcal{J}$ and finally to robot $j_k^{\overline{L}_k}$, which changes its succeeding robot to j_k^1. Through this procedure, the robots in $\mathcal{I}$ transit to the relay mode for the assistance task, while the robots in $\mathcal{J}$ remain in the spread mode for exploration.

2) Transition From Relay Mode to Spread Mode: Upon completing the assistance task, relay robots transition back to spread mode. This is triggered when the messenger returns with updated data at $t_{\mathrm{r}} > 0$, obtains the remaining task time $t_{\mathrm{w}} > 0$ from the operator, and checks the condition:

$$t_{\mathrm{r}} + t_{\mathrm{w}} + \max_{\ell,j}\left\{T_{\texttt{Nav}}(p_k^\ell, p_j^\star)\right\} \leqslant \max_{j}\{t_j^\star\}, \tag{8}$$

where $t_j^\star$ and $p_j^\star$ denote the *last* planned meeting time and location by the robots in the spread mode. The above condition ensures that any relay robot can reach at least one planned communication event. Afterwards, the transition from relay mode to spread mode proceeds as follows: (I) The earliest communication event $c_{j^\star}$ that satisfies (8) is determined, with $c_{j^\star+1}$ being the subsequent event in the spread mode. A collision-free path from $p_{j^\star}$ to $p_{j^\star+1}$ is then computed as $\mathbf{p}_{j^\star}$; (II) Each relay robot $i \in \mathcal{I}$ finds the nearest point $\widehat{p}_i \in \mathbf{p}_{j^\star}$ and computes its arrival time $\widehat{t}_i \triangleq t_{\mathrm{r}} + t_{\mathrm{w}} + T_{\texttt{Nav}}(p_i, \widehat{p}_i)$. Thus, the relay robots are re-ordered by their arrival time to $\mathbf{p}_{j^\star}$ as $i_0 i_1 \cdots i_{L_k}$; (III) Each relay robot i_ℓ communicates with its predecessor $i_{\ell-1}$ at event $c(i_{\ell-1}, i_\ell)$ and with its successor $i_{\ell+1}$ at event $c(i_\ell, i_{\ell+1})$, $\forall \ell = 1, \cdots, L_{k-1}$. Moreover, robot i_0 communicates with its predecessor $j^\star$ at event $c(j^\star, i_0)$ and with its successor i_1 at event $c(i_0, i_1)$; robot i_{L_k} communicates with its predecessor $i_{L_{k-1}}$ at event $c(i_{L_{k-1}}, i_{L_k})$ and with its successor $j^\star + 1$ at event $c(i_{L_k}, j^\star + 1)$. In this way, robot i_{L_k} effectively replaces the original predecessor of robot $j^\star + 1$. The communication events are then defined as follows: $c(i_{\ell-1}, i_\ell) = (\widehat{t}_{i_\ell}, \widehat{p}_{i_\ell})$, $c(j^\star, i_0) = (\widehat{t}_{i_0}, \widehat{p}_{i_0})$, and $c(i_{L_k}, j^\star + 1) = c_{j^\star+1}$. Through this procedure, the planned communication events are updated, by which the robots in $\mathcal{I}$ can transit back to the spread mode.

D. Online Execution and Adaptation

As summarized in Alg.1, the execution alternates between spread mode, return events, and relay mode. Initially, all robots follow the spread mode for exploration and communication from Sec.III-A, which governs inter-robot communication and return events. Upon receiving assistance requests, relay topology and roles are optimized via the hybrid method in Sec.III-B, and assigned relay robots transition according to Sec.III-C. After completing a task, they return to spread mode unless another task is queued, in which case they navigate directly to the next without reverting. The operator prioritizes tasks based on ρ_k when multiple requests arrive.

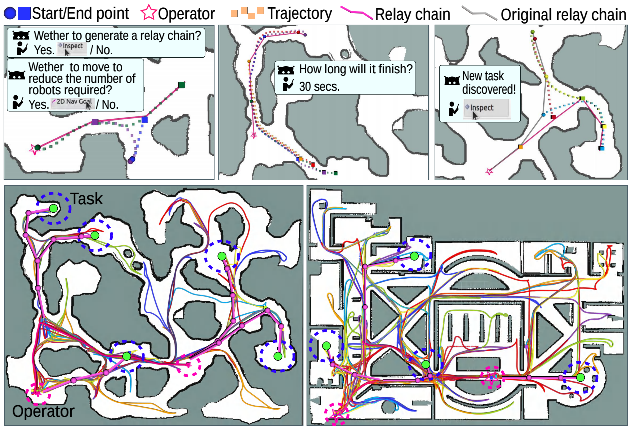

Fig. 4: Representative simulation results. Online interactions between the operator and the fleet during three requests (**Top**). Fully-explored cave and emporium with robots and operator trajectories, where green dots indicate assistance tasks and purple dots indicate relay robots.(**Bottom**).

IV. NUMERICAL EXPERIMENTS

For further validation, numerical simulations and hardware experiments are presented in this section. The proposed method is implemented in `Python3` within the framework of `ROS`, and tested on a computer with an Intel Core i7-13700KF CPU. Simulation and experiment videos can be found in the supplementary files.

A. System Description

The robotic fleet consists of 8 differential-driven UGVs, which are simulated in the `Stage` simulator and visualized in the `Rviz` interface. As shown in Fig. 4, two different workspaces are tested: (I) a large subterranean cave of size $65m \times 57m$ with numerous tunnels; (II) a large emporium of size $60m \times 50m$ with many connected rooms. The occupancy grid map [16] is adopted with a resolution of $0.1m$ and a sensor range of $10m$, generated via the `gmapping` SLAM package. Each robot navigates using the navigation stack `move_base`, with a maximum linear velocity of $0.6m/s$ and angular velocity of $1.5rad/s$. The operator could move with a velocity of $0.3m/s$. The robots and the operator start initially from the bottom-left corner of the map.

Moreover, two robots can only communicate if they have a line of sight (LOS) and are within a range of $5m$, the same between robots and the operator. The periodic update by T_{h} to the operator is set to $60s$. As shown in Fig. 4, five (Cave) or four (Emporium) assistance tasks are initially unknown, and discovered by robots gradually. Lastly, the operator interacts with the robotic fleet through the terminal and a customized GUI in `Rviz`.

B. Simulation Results

1) Subterranean Cave: As shown in Fig. 5, the fleet fully explores the 1903 square meters within $1498s$, while the operator receives the full map in $1571s$. Moreover, during exploration, **five** assistance tasks are received and completed, the last of which at $1548s$. Assistance task stages are recorded as in Fig. 5: T_1: time from discovery to operator

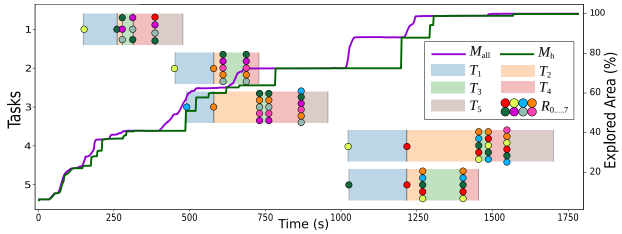

Fig. 5: Progress of exploration and task fulfillment by 8 robots in the cave environment. Participants of five stages of each assistance task are indicated by filled dots; the map of the operator M_{h} follows the union of robot maps $M_{\texttt{all}}$.

reception; T_2: operator interaction; T_3: transition from spread mode to relay mode; T_4: task execution via chain (T_k); T_5: transition from relay mode to spread mode. Fig. 4 shows the T_3 of Task 1, the T_5 of Task 3, the directly transition in relay mode from Task 5 to 4. Note that the participants of each assistance task are different and several tasks are fulfilled in parallel. It is worth emphasizing that the average transition time (including from and back to spread mode) for all tasks is around $130.8s$. Lastly, via the proposed scheme of direct transition among assistance tasks, robots 3-7 transit from Task 2 to 3 within only $32s$, which is significantly shorter than the average duration of $64s$. Similar phenomenon occurs during the concurrent execution of Tasks 4 and 5.

2) Emporium: For the emporium environment, the structure of map is more complicated for exploring, as shown in Fig. 4, in total **four** assistance tasks are imposed and completed in $1536s$. Tasks 1 and 2 are completed via direct transition at $474s$, with a transition time of $40s$. Tasks 3 and 4 are completed at $1020s$ and $1500s$, respectively. In tasks 3 and 4, the operator choose to move along the relay topology, reducing the number of relay robots by 2 and 5, respectively, allowing more robots to continue exploring the map. Finally, the fleet fully explores the 1737 square meters within $1591s$, and the complete map is obtained by the operator at $1620s$.

C. Comparison

The proposed framework **FlyKites** is compared against four baselines: (I) **SEP** separates exploration and task execution, handling all requests post-exploration; (II) **OP-DM**, where operator directly moves to the requests without a chain; (III) **OP-STA**, where the operator stays static; and (IV) **NO-DT**, where the direct transition among tasks is not allowed. The compared metrics are the completion rate and time of all tasks, and the average and maximum latency of all tasks, where task latency refers to the duration of time between observing a task and finishing it.

Table I summarizes the results of multiple simulation runs. It can be seen that **FlyKites** achieves the best performance over all baselines for all metrics. Although **SEP** completes tasks with only slightly longer time than **FlyKites**, it suffers from much larger task latency due to separation of exploration and task execution. Without the proposed chain of communication, the completion time of **OP-DM** is almost twice that of ours ($3105s$ vs. $1571s$). Note that **NO-DT** has also a long time ($2792s$), as significant time is spent on mode transition when there are multiple tasks. Lastly, if operator stays static as in **OP-STA**, only 60% tasks can be handled due to the inadequate number of robots to form a chain for Tasks 4 and 5. In contrast, our method requires the minimal duration to accomplish all tasks with the minimum latency.

TABLE I: Comparison with Baselines.

Method	Task Completion [%]	Finish Time [s]	Avg. Task Latency [s]	Max. Task Latency [s]
FlyKites	**100**	**1571**	**303.6**	**465.0**
SEP	100	1647	894.0	1055.0
OP-DM	100	3105	423.6	743.0
OP-STA	60	\	∞	∞
NO-DT	100	2792	425.6	676.0

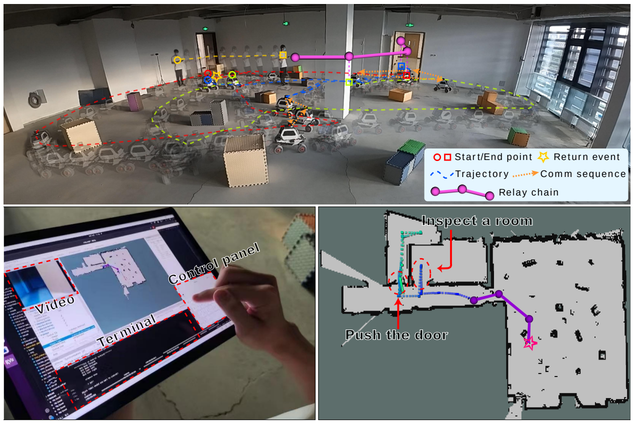

Fig. 6: Hardware experiments where the operator interacts online with the fleet via a tablet and ad-hoc communication network. A communication chain is formed for the second task via three UGVs for video stream and tele-operation.

D. Hardware Experiments

As shown in Fig. 1 and 6, an operator deploys 3 ground robots to explore an office environment of size $35m \times 20m$, and interacts with the fleet with a tablet. Each robot and the operator are equipped with an ad-hoc network device for close-range communication (AP-DLINK1402A). The office is fully explored at $107s$, after which two robots explore the corridor and the third one returns to operator. A communication chain of three robots is then formed at $294s$, for which the operator moves along the chain to reduce the number of relays from 4 to 3. Then, the operator controls the end robot to (I) inspect a room; and (II) push the door of another room to fully explore the environment. Finally, the two requests are accomplished at $523s$. It is worth noting that via the live video stream from the end robot, the operator can control it smoothly and safely **without** direct line of sight.

V. CONCLUSION

This work proposes a novel and generic framework (FlyKites) for the online interaction and assistance between a dynamic operator and a robotic fleet in unknown and communication-constrained environments. Future work will focus on the integration with other collaborative tasks besides exploration, e.g., search and rescue.

REFERENCES

[1] Jacopo Banfi, Alberto Quattrini Li, Nicola Basilico, and Francesco Amigoni. Communication-constrained multirobot exploration: Short taxonomy and comparative results. In *Proceedings of the IROS workshop on on-line decision-making in multi-robot coordination (DEMUR2015)*, pages 1–8, 2015.

[2] Wolfram Burgard, Mark Moors, Cyrill Stachniss, and Frank E Schneider. Coordinated multi-robot exploration. *IEEE Transactions on robotics*, 21(3):376–386, 2005.

[3] Hong Cai and Yasamin Mostofi. Human-robot collaborative site inspection under resource constraints. *IEEE Transactions on Robotics*, 35(1):200–215, 2019.

[4] Kyle Cesare, Ryan Skeele, Soo-Hyun Yoo, Yawei Zhang, and Geoffrey Hollinger. Multi-uav exploration with limited communication and battery. In *IEEE international conference on robotics and automation (ICRA)*, pages 2230–2235, 2015.

[5] Rafael Gonçalves Colares and Luiz Chaimowicz. The next frontier: Combining information gain and distance cost for decentralized multi-robot exploration. In *Annual ACM Symposium on Applied Computing*, pages 268–274, 2016.

[6] Micael Santos Couceiro. An overview of swarm robotics for search and rescue applications. *Artificial Intelligence: Concepts, Methodologies, Tools, and Applications*, pages 1522–1561, 2017.

[7] Joel M Esposito and Thomas W Dunbar. Maintaining wireless connectivity constraints for swarms in the presence of obstacles. In *IEEE International Conference on Robotics and Automation*, pages 946–951, 2006.

[8] Yuman Gao, Yingjian Wang, Xingguang Zhong, Tiankai Yang, Mingyang Wang, Zhixiong Xu, Yongchao Wang, Yi Lin, Chao Xu, and Fei Gao. Meeting-merging-mission: A multi-robot coordinate framework for large-scale communication-limited exploration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 13700–13707, 2022.

[9] Meng Guo and Michael M Zavlanos. Multirobot data gathering under buffer constraints and intermittent communication. *IEEE Transactions on Robotics*, 34(4):1082–1097, 2018.

[10] Dirk Holz, Nicola Basilico, Francesco Amigoni, and Sven Behnke. Evaluating the efficiency of frontier-based exploration strategies. In *International Symposium on Robotics*, pages 1–8. VDE, 2010.

[11] Ahmed Hussein, Mohamed Adel, Mohamed Bakr, Omar M Shehata, and Alaa Khamis. Multi-robot task allocation for search and rescue missions. In *Journal of Physics: Conference Series*, volume 570, page 052006, 2014.

[12] Filip Klaesson, Petter Nilsson, Tiago Stegun Vaquero, Scott Tepsuporn, Aaron D Ames, and Richard M Murray. Planning and optimization for multi-robot planetary cave exploration under intermittent connectivity constraints. 2020.

[13] Yaroslav Marchukov and Luis Montano. Fast and scalable multi-robot deployment planning under connectivity constraints. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 1–7, 2019.

[14] Yaroslav Marchukov and Luis Montano. Fast and scalable multi-robot deployment planning under connectivity constraints. In *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 1–7, 2019.

[15] Ryan J Marcotte, Xipeng Wang, Dhanvin Mehta, and Edwin Olson. Optimizing multi-robot communication under bandwidth constraints. *Autonomous Robots*, 44(1):43–55, 2020.

[16] Hans Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 116–121, 1985.

[17] R.R. Murphy. Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2):138–153, 2004.

[18] Indraneel Patil, Rachel Zheng, Charvi Gupta, Jaekyung Song, Narendar Sriram, and Katia Sycara. Graph-based simultaneous coverage and exploration planning for fast multi-robot search. *arXiv preprint arXiv:2303.02259*, 2023.

[19] Yuanteng Pei, Matt W Mutka, and Ning Xi. Coordinated multi-robot real-time exploration with connectivity and bandwidth awareness. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5460–5465, 2010.

[20] Yuanteng Pei, Matt W Mutka, and Ning Xi. Connectivity and bandwidth-aware real-time exploration in mobile robot networks. *Wireless Communications and Mobile Computing*, 13(9):847–863, 2013.

[21] Pavel Petráček, Vít Krátkỳ, Matěj Petrlík, Tomáš Báča, Radim Kratochvíl, and Martin Saska. Large-scale exploration of cave environments by unmanned aerial vehicles. *IEEE Robotics and Automation Letters*, 6(4):7596–7603, 2021.

[22] Gregg Podnar, John M. Dolan, Alberto Elfes, Marcel Bergerman, H. Benjamin Brown, and Alan D. Guisewite. Human telesupervision of a fleet of autonomous robots for safe and efficient space exploration. In *IEEE/ACM International Conference on Human-Robot Interaction*, 2006.

[23] Christopher Reardon, Kevin Lee, John G Rogers, and Jonathan Fink. Communicating via augmented reality for human-robot teaming in field environments. In *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 94–101. IEEE, 2019.

[24] Martijn N Rooker and Andreas Birk. Multi-robot exploration under the constraints of wireless networking. *Control Engineering Practice*, 15(4):435–445, 2007.

[25] Maira Saboia, Lillian Clark, Vivek Thangavelu, Jeffrey A Edlund, Kyohei Otsu, Gustavo J Correa, Vivek Shankar Varadharajan, Angel Santamaria-Navarro, Thomas Touma, Amanda Bouman, et al. Achord: Communication-aware multi-robot coordination with intermittent connectivity. *IEEE Robotics and Automation Letters*, 7(4):10184–10191, 2022.

[26] Matthew A Schack, John G Rogers, and Neil T Dantam. The sound of silence: Exploiting information from the lack of communication. *IEEE Robotics and Automation Letters*, 2024.

[27] Zhuoli Tian, Yuyang Zhang, Jinsheng Wei, and Meng Guo. ihero: Interactive human-oriented exploration and supervision under scarce communication. In *Robotics: Science and systems (RSS), Delft, Netherlands*. IEEE, 2024.

[28] Tiago Vaquero, Martina Troesch, and Steve Chien. An approach for autonomous multi-rover collaboration for mars cave exploration: Preliminary results. In *International Symposium on Artificial Intelligence, Robotics, and Automation in Space (SAIRAS)*, 2018.

[29] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pages 146–151, 1997.

[30] Brian Yamauchi. Decentralized coordination for multirobot exploration. *Robotics and Autonomous Systems*, 29(2-3):111–118, 1999.

[31] Michael M Zavlanos, Magnus B Egerstedt, and George J Pappas. Graph-theoretic connectivity control of mobile robot networks. *Proceedings of the IEEE*, 99(9):1525–1540, 2011.

[32] Boyu Zhou, Hao Xu, and Shaojie Shen. Racer: Rapid collaborative exploration with a decentralized multi-uav system. *IEEE Transactions on Robotics*, 2023.